STATISTICAL METHODS FOR FEATURE EXTRACTION IN SHAPE ANALYSIS AND BIOINFORMATICS

A Dissertation
Presented to
The Academic Faculty

by

Xavier Jean Maurice Le Faucheur

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Electrical and Computer Engineering

Georgia Institute of Technology
May 2010

STATISTICAL METHODS FOR FEATURE EXTRACTION IN SHAPE ANALYSIS AND BIOINFORMATICS

Approved by:

Professor Allen Tannenbaum, Advisor
School of Electrical and Computer Engineering
Georgia Institute of Technology

Professor Anthony Yezzi
School of Electrical and Computer Engineering
Georgia Institute of Technology

Professor Brani Vidakovic, Advisor
Department of Biomedical Engineering
Georgia Institute of Technology

Professor David Anderson
School of Electrical and Computer Engineering
Georgia Institute of Technology

Professor Jeff Shamma
School of Electrical and Computer Engineering
Georgia Institute of Technology

Professor Rina Tannenbaum
School of Materials Science and Engineering
Georgia Institute of Technology

Date Approved: February 26, 2010

To my parents, Jean and Catherine,

and my grand-parents,

Henri and Micheline Le Faucheur,

Maurice and Geneviève Faguet.

ACKNOWLEDGEMENTS

During my time as a Ph.D. student at the Georgia Institute of Technology I have been fortunate to meet, work and exchange with amazing people. Your all have made my time here fantastic. In particular, I would like to thank:

My Advisor. Professor Allen Tannenbaum, your wisdom, your knowledge, and your humor have been truly inspiring. It has been an absolute honor and a real pleasure to work under your supervision. Thank you for your guidance and friendship over the years.

My Co-advisor. Professor Brani Vidakovic, your knowledge and your kindness have been tremendously supportive. Thank you for having taught me about statistics and helped me go through my PhD research.

My Mentor and collaborator. Doctor Eli Hershkovits, you pulled me into the dark side of bioinformatics and taught me about RNA. Thank you for your kindness, your help and your patience.

My Thesis Committee. Professor Rina Tannenbaum, with whom I had the pleasure to work; Professors Anthony Yezzi, David Anderson, and Jeff Shamma whose remarks helped guide and improve my thesis.

Mentors in the Lab. Delphine Nain, Samuel Dambreville, and Eric Pichon. I am honored to have had you as friends and lab-mates and to have learned from you during that time.

My AT-gang mates. Shawn Lankton (AT #34) and Romeil Sandhu (AT #36). Thank you for having initiated me to the American culture, for all the fun we have had in and outside the lab, and, of course, for that fantastic friendship we have built over these past few years. You are my favorite southern boys and I look forward to the many more adventures to come.

Members of the MINERVA Lab. Yi, Behnood, Pete, Ivan, Jehoon, Vandana, Jimi, Gallagher, Tauseef, Ponnappan, John, Marc, Patricio, Yogesh, Yan, thank you all for making our lab a friendly, scholarly, and wonderful work place.

Van Leer VIPs. Marilouise Mycko, Jacqueline Trappier, Tasha Torrence, and Christopher Malbrue. Thank y'all for your help, your collaboration, and your kindness.

My friends. Julien, Laurent, Arnaud, Nicolas, Philip, Vincent, Emilie, Benoit, Antoine, Juan, Kevin, Ludvic, Joe, Geneviève, the "French connection" in Atlanta, my GT volleyball team, my friends from SFX, Sainte Geneviève, Supélec, my friends from Atlanta or elsewhere... Your support and your friendship have been invaluable. Thank you all for always being there for me.

My Family. Mom, Dad, Anne, Carole and Isabelle. You gave me so much, and helped me become who I am today. Thank you for everything.

Lucie. Your Love and support has been the most beautiful gift. Thank you for sharing this journey with me. I am grateful for everything we have built and accomplished together.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

SUMMARY

Feature extraction aims to explain the underlying phenomena of interest of a given set of input data by simplifying the amount of resources required to accurately describe it. This terminology remains very broad as it refers to a lot of different objectives and encompasses multiple types of techniques, methods and processes.

The work contained in this thesis explores two types of feature extraction, from two different domains, namely 3D shape analysis and bioinformatics. The objective of both projects is to detect and understand the relevant information from a noise corrupted data set. However, the two processes significantly differ from each other, as one aims to compress and smooth signals while the other consists of clustering data.

In the first part of this thesis, a method for shape representation, compression and smoothing is proposed. First, it is shown that, similarly to spherical shapes, triangulated genus-one surfaces can be encoded using second generation wavelet decomposition. Next, a novel model is proposed for wavelet-based surface compression and smoothing. This part of the work aims to develop an efficient and robust process for eliminating irrelevant and noise-corrupted parts of the shape signal. Surfaces are encoded using wavelet filtering, and the objective of the proposed methodology is to separate noise-like wavelet coefficients from those contributing to the relevant part of the signal. The technique developed in this thesis consists of adaptively thresholding coefficients using a data-driven Bayesian framework. Once "thresholding" is performed, the coefficients that have been identified as irrelevant are removed and the inverse wavelet transform is applied to the "clean" set of wavelet coefficients. Experimental results show the efficiency of the proposed technique for surface smoothing and compression.

The second part of this thesis proposes a statistical model for studying RNA (RiboNucleic Acid) spatial conformations. The functional diversity of the RNA molecule depends on the ability of the RNA polymer to fold into a large number of precisely defined spatial forms. Therefore, one of the main challenges of bioinformatics is to establish a clearer understanding of the structure/function relationships in these molecules. If the functionality of a specific substructure (or *unit block*) from a given part of a RNA strand is known, then the functionality of similar substructures is assumed to be similar. Therefore, it is important to find an efficient way to classify the unit blocks of the RNA molecule. Each type of substructure can be geometrically characterized by a set of d parameters, which defines the spatial arrangement of its constituents. Thus, a set of substructures from the same family can be represented as a point cloud in a d-dimensional data space. A similarity measure can therefore be defined to perform clustering on this given data set and classify the corresponding substructures into a limited number of groups. In the proposed work, a statistical clustering model is applied to this RNA structure classification problem. First, single nucleotide structures are classified with respect to their spatial configurations. Application of the method to various data sets validates the process and further analysis is conducted to compare the results to other classifications. Second, the same clustering scheme is applied to base doublet geometries (base pairs and base stacking). These conformations offer more complex and challenging data sets. The proposed clustering results bring new features into the existing classification schemes.

CHAPTER I

INTRODUCTION

This chapter first explains the notions of feature extraction, data de-noising and dimensionality reduction in noise-corrupted data. Next, this introduction will allow the reader to understand how the work presented in this dissertation relates to these three concepts and will list the major contributions of the proposed work. Finally, the content of each subsequent chapter is described to help the reader understand the structure of the remainder of the dissertation.

1.1 On the Notion of Feature Extraction

Data analysis may generally be described as a three-step process, as presented in Figure 1.1. First, data is usually collected either "manually" or in an automated manner. The observed data is referred to as *raw data*. Next, *feature extraction* aims to explain the underlying phenomena of interest of the set of raw data by simplifying the amount of resources required to accurately describe it. In various fields, such as image processing or bio-informatics, raw data is corrupted with irrelevant and undesired variations, or *noise*, that are meant to be discarded. Thus, feature extraction methods usually are combinations of noise removal (also called *de-noising*), structure detection, and dimensionality reduction techniques. In general, an optimal balance needs to be found between fineness and complexity of the extracted features. The output should use a minimal amount of resources while being able to accurately describe the underlying phenomena of interest of the data. Once the relevant part of the signal has been extracted, detailed analysis may be conducted, hypotheses may be drawn, and further applications may be considered by the end-user.

Figure 1.1: The general three-step data analysis process.

This dissertation focuses on the feature extraction part of the data analysis process.

In multiple fields, the extraction of noise-free features from an original data set constitutes a major processing step and provides a base for further analysis. This type of problem can be decomposed along two axes. The first challenge consists of quantifying the goodness of the extracted features, i.e. in validating the significance of the output. Based on this criterion, one can distinguish between two different contexts: those where results are quantifiable and those where results are non-quantifiable. In the former case, full or partial validation is made possible by the existence of referenced classifications or ground truth, whereas, in the latter, comparison is only potentially made between different techniques, but no unique reference can be used. Therefore, depending on whether or not one has access to some sort of objective validation, the significance of the results becomes more or less subjective. In addition to this validation issue, distinction can be made between data sets for which prior knowledge on the underlying structure is available (and/or used) and those for which there is none. For example, prior knowledge may include information on the geometry of the underlying signal, on the number of data subsets or underlying clusters, or on the total number of outlying and irrelevant data points. These two dimensions being defined, it is possible to categorize case scenarios using a 2×2 matrix, which

	Validation	**No Validation**
Prior Knowledge	CASE 1	CASE 2 (*e.g. shape denoising*)
No Prior Knowledge	CASE 3 (*e.g. RNA clustering*)	CASE 4

Figure 1.2: Prior Knowledge - Validation Matrix: feature extraction contexts can be classified along two dimensions. Four scenarios are defined. Case 1 is the simplest type of scenario and Case 4 is usually the most difficult situation one may encounter. Cases 2 and 3 are illustrated by the two examples developed in the proposed research.

is referred to as the *Prior Knowledge - Validation Matrix* (see Figure 1.2).

The work presented in this thesis aims to illustrate two of the four scenarios presented in Figure 1.2. Here is an introductory description of these two examples:

1- The first part of the proposed research focuses on developing a wavelet-based method for 3D shape denoising and compression. Noise-corrupted surfaces constitute the input of the system, and the objective of this work consists of "extracting" smooth and compressed versions of the original shapes. Quantitative validation of the results is very rarely available. Indeed, although compression rates may be easily computed, smoothing performance remains a very subjective concept to evaluate. However, prior knowledge on the resolution of the representation, on the nature of the shape and on the characteristics of the tools used to encode the signal is usually available to the user. This prior information enables one to customize the model by making it more adaptive. Therefore, this topic fits in case #2 of the aforementioned matrix (Figure 1.2).

2- The second part proposes using a non-parametric clustering method for classifying RNA (RiboNucleic Acid) conformations. The local conformation of RNA

molecules is an important factor in determining their catalytic and binding properties. Three types of conformations are taken into account: single residue, base pair geometry, base stacking geometry. Each type of RNA conformation is characterized by a set of parameters that define the spatial arrangement of its constituents, i.e. the degrees of freedom of the RNA sub-structure (torsion angles, base-to-base distances, base-to-base rotation angles). Classification in the conformation space consists in finding clusters of similar sub-structures. Automated non-parametric clustering methods offer the possibility to obtain such a classification without any prior knowledge on the number and/or size of clusters. However, the application of these techniques to well-documented conformations, references to RNA data bases, and the analysis of the content or chemical properties of each cluster's elements all constitute potential bases for validation and explanation of the resulting clustering. Therefore, this context fits in case #3 of the 2×2 matrix (Figure 1.2).

1.2 Contributions and Organization of this Thesis

Within the scope of the first "project", this thesis proposes two major contributions:

- **A novel model for multi-scale representation of genus-one surfaces using second generation wavelets.** This extends the work of Nain *et al.* [74] that proposed a wavelet-based approach to accurately encode spherical surfaces. This extension is motivated by the need to develop a common framework for multi-scale shape analysis. Using second generation wavelets [87], any signal defined on a multi-resolution mesh can be efficiently decomposed in scale and space and represented by a limited set of large coefficients. In the proposed work, we first develop an algorithm to equip genus-one surfaces with a multi-resolution grid. Next, we show how second generation wavelets can be built on a toroidal multi-resolution mesh. The encoding of surfaces by projection onto these wavelet functions is then described. Finally, experimental results

illustrate the application of the proposed algorithm on genus-one surfaces.

- **A novel statistical wavelet shrinkage framework for surface de-noising and compression.** Surfaces are encoded using second generation wavelets, and the proposed model aims to efficiently remove noise-like wavelet coefficients. The two statistical *shrinkage* models described in this work offer efficient frameworks to threshold coefficients and eliminate those that are considered as irrelevant parts of the signal. In the context of surface encoding, adaptive Bayesian shrinkage rules provide interesting features for signal compression and smoothing. Their efficiency is mainly due to their capability for incorporating local information into their framework. This local information brings interesting value to the model as it characterizes the spatial neighborhood of the node at which a coefficient is defined. If the only amplitude of a coefficient was used in the thresholding decision process, the shrinkage rule would not lead to consistent smoothing or compression in the local region of the coefficient. Thus, in the proposed frameworks, parent coefficients from coarser levels, as well as local curvature of the surface, are taken into account for the thresholding process.

Within the second part of this thesis, two major contributions are made:

- **A clustering methodology for studying RNA conformations.** The local conformation of RNA molecules is an important factor in determining their catalytic and binding properties. The analysis of such conformations is particularly difficult due to the large number of degrees of freedom, such as the measured torsion angles per residue and the inter-atomic distances among interacting residues. In order to understand and analyze the structural variability of RNA molecules, this work proposes a methodology for detecting repetitive conformational sub-structures along RNA strands. Clusters of similar structures in the conformational space are obtained using a nearest-neighbor search

method based on the statistical mechanical Potts model. The proposed technique is mostly automatic and may be applied to problems where there is no prior knowledge on the structure of the data space, in contrast to many other clustering techniques. Clustering was performed on two types of RNA substructures: backbone conformations and base-base geometries. First, results are reported for both single residue conformations, where the parameter set of the data space includes four to seven torsional angles. Next, for the case of base pair geometries, the data space was reduced to two dimensions and the Potts model was applied to various data sets. For these two cases, a very good match between the results of the proposed clustering method and existing classifications was observed, with only few exceptions.

- **A new classification for base stacking geometries.** Base stacking interactions are base-base interactions where one base is located above the other. By applying the aforementioned Potts model clustering to the problem of base stacking geometries, we were able to deliver a new classification for this category of RNA sub-structures. New results are reported, the content and the geometry of the new clusters are discussed and different ways to validate these results are presented.

These contributions will be organized as follows:

- **Chapter 2:** Describes the proposed methodology for the encoding of genus-one surfaces using second generation wavelets. Particular attention is given to the explanation of the re-meshing process.

- **Chapter 3:** Presents the statistical wavelet thresholding framework for 3D shape denoising and compression. Two mathematical models are derived and experimental results are shown.

- **Chapter 4:** Describes the application of the Potts model clustering to the problem of RNA conformations. After presentation of the model and comparison to other clustering techniques, a detailed analysis of the results is conducted.

- **Chapter 5:** Offers a summary of the presented work, draws conclusions from the thesis, and discusses possible directions for future work.

CHAPTER II

WAVELET ANALYSIS FOR GENUS-ONE SURFACES

Various formats exist for encoding 3D objects. In many applications, 3D objects are only represented by their boundaries, i.e. by their external surface. Parameterized surfaces and polygonal surfaces (or polygonal meshes) constitute the two main families of surface representations. For visualization, transmission and modeling purposes, polygonal meshes have become a very useful and commonly used tool in computer graphics. A polygonal mesh (F, E, V) is defined as a collection of vertices (V), edges (E), and faces (F). The polygonal faces approximate the original surface, two adjacent faces share a common edge, and each edge connects two vertices. Each vertex is a sampled point of the surface. In the proposed work, surfaces are equipped with triangulated meshes.

When using mesh structures, double encoding is required to build, transmit, and store surfaces. First, the spatial coordinates of the vertices encode the shape of the surface. This is called *data encoding*. Second, encoding the mesh structure itself allows one to know which triangles connect which vertices. This is called *connectivity encoding*. Various techniques may be used to implement this double encoding, and among these methods, multi-scale schemes offer nice settings and advantages as they provide a compact and well-organized way to encode connectivity. The structure of a multi-scale mesh consists of a nested grid that can be decomposed into several *levels of resolution*. Such a mesh is built recursively by successively adding new vertices to an initial coarse structure, where each new subset of nodes adds a finer resolution level to the existing mesh. Vertices are therefore classified by resolution, and the connectivity between two successive levels follows a simple and regular construction

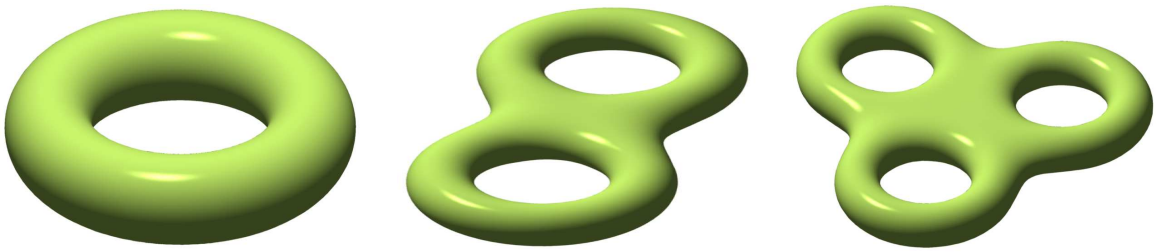
scheme (e.g. a new vertex is placed at the middle of an edge defined by two vertices from the precedent level).

Different approaches have been proposed for creating multi-scale representations of 3D meshes, using *second generation wavelets*. Second generation wavelets are adaptive wavelets that can be built on manifolds with non-regular grids. Lounsbery *et al.* [66] first proposed a framework for representing surfaces using adaptive wavelet basis functions. This inspired the work of Schröder *et al.* [87], who developed a way to efficiently encode any function defined on a 3D multi-resolution mesh. This method facilitates the incorporation of custom features and offers a large range of potential applications.

Nain *et al.* [74] showed that spherical wavelets [87] were offering very efficient properties for multi-scale analysis of triangulated spherical surfaces. Indeed, this type of multi-scale decomposition provides a way to compress the relevant part of the signal into a reduced set of coefficients. Furthermore, wavelets offer a very efficient scheme for customized compression and smoothing (see Chapter 3). Finally, the use of a multi-scale decomposition tends to simplify the mesh connectivity encoding. Surfaces are re-triangulated and equipped with a multi-resolution mesh, in which each level of resolution is easily recoverable from the precedent one. This re-meshing process, if executed in a lossless manner, is aligned with the objective of compression for storage and transmission.

The work of Nain *et al.* shows that second generation wavelet decomposition works well for surfaces with a spherical topology. The work presented in this chapter is motivated by the possible extension of this wavelet encoding scheme to more complex topologies. More specifically, a framework for multi-scale analysis will be proposed for genus-one surfaces.

This chapter presents a framework that allows genus-one shapes (i.e. non-spherical shapes with one hole) to be encoded using second generation wavelets. In the next



(a) (b) (c)

Figure 2.3: Shapes with different genuses. (a): Genus 1, (b): Genus 2, (c): Genus 3.

section, the need for this type of analysis is motivated, the nature of the wavelet functions used in this work is described and explanations on the way these may be defined on a genus-one mesh are given. In Section 2.3, after reviewing the major existing techniques for surface re-meshing through a brief literature survey, a description of the proposed methodology for re-triangulating surfaces is given. Next, Section 2.4 presents some experimental results on the proposed surface encoding. Finally, concluding remarks are made in Section 2.5.

2.1 Motivation for Genus-One Surface Analysis

The topology of a surface is a commonly used criterion for 3D geometry classification. Two surfaces are *topologically equivalent* if there exists a homeomorphism that transforms one to the other and vice-versa. The topology of a surface is characterized by its genus. The genus of a surface is an integer representing the maximum number of cuttings along non-intersecting closed simple curves without rendering the resultant manifold disconnected. This is concretely equal to the number of *handles* on it (Figure 2.3).

In the world of computer graphics, shapes with different topologies are manipulated, stored, analyzed. Moreover, surfaces with a genus equal to or greater than one usually exhibit more complex shape features, such as very high curvature regions, which are worth analyzing in details. These features may require further work than

what has been done for spherical shapes.

The number of handles characterizes the genus of a surface. Thus, one may consider that genus-one shapes constitute unit building blocks for the analysis of more complex shapes. Therefore, in the work presented in this chapter, analysis has been conducted on genus-one shapes only. More specifically, a wavelet analysis scheme, similar to that proposed in [74], has been developed to encode the spatial coordinates of triangulated genus-one surfaces.

2.2 Second Generation Wavelets on Genus-One Meshes

In the case of spherical surfaces, Nain *et al.* showed that the coordinates of a 3D mesh could be efficiently decomposed in space and scale by projecting it onto a set of spherical wavelets. These wavelets, developed by Schröder *et al.*, are biorthogonal basis functions that one may build on any spherical multi-resolution (e.g. a multi-scale subdivision of the sphere). The work presented here proposes building a similar set of basis functions on a multi-resolution mesh that exhibits a genus-one topology. Since genus-one surfaces are topologically equivalent to a single torus, the wavelet basis functions will be defined on a multi-resolution subdivision of a torus.

In this section, a description of the second generation wavelets that are used in this work to encode the surfaces is given. It is also shown how wavelet basis functions can be built on a grid with a torus structure.

2.2.1 Second Generation Wavelet Scheme

A spherical wavelet basis is an L^2 basis composed of functions defined on a sphere that are localized in space and scale. The basis consists of scaling functions defined at the coarsest scale and wavelet functions defined at subsequent scales. The wavelets used in this work are similar to the discrete biorthogonal spherical wavelets developed by Schröder *et al.* [87]. These *second generation wavelets* have very nice properties and settings (finite support, fast transform, opportunities for customization, *lifted bases*)

and can be developed on manifolds with non-regular grids. The major difference with the classical wavelets is that the "filter" changes in space in order to reflect the variations in the surface and its measure. The size of the support of two basis functions from the same level of resolution may vary. This means that wavelet functions defined on a mesh are not scaled and shifted versions of the function on a coarser grid.

As presented in [87], a *lifting scheme* is used to build fully biorthogonal wavelet bases. A *dual lifting* may also be used in order to increase the number of vanishing moments of the dual wavelets, which has the effect to achieve higher compression (see discussion in Section 2.4.2).

Construction of a multi-resolution mesh on a torus: As mentioned earlier, the second generation wavelets that will need to be used here are defined on surfaces that are topologically equivalent to a torus and equipped with a multi-resolution mesh. Therefore, this mesh can be built by recursively subdividing an initial coarse triangulated torus (or planar rectangle which corresponds to a torus that would have been cut and unfolded). After j subdivisions, the refined grid contains $K(j)$ nodes. The $j+1^{th}$ subdivision introduces new nodes, which can be denoted by an index set $M(j)$. These new nodes are placed at the middle of each existing edge. Therefore, a subdivision splits each existing triangle into 4 new triangles. The complete set of nodes at the $j+1^{th}$ level of resolution is given by $K(j+1) = K(j) \bigcup M(j)$. Thus, a multi-resolution structure is built that will enable the multi-scale decomposition of the signal and the development of wavelet bases. Figure 2.4 shows successive subdivisions of the mesh.

Scaling and wavelet functions:

Given this multi-resolution grid, one may now build the basis functions onto which the signal will be projected and that will help encode the signal at level of resolution. A scaling function $\varphi_{j,k}$ at a chosen resolution j is defined as a scalar function defined on a mesh of resolution j and centered at node k. The set of scaling functions is

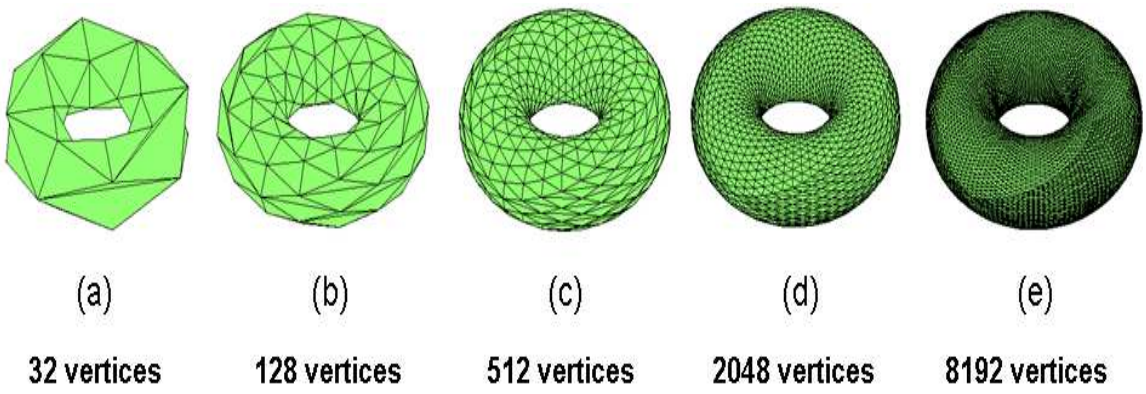


Figure 2.4: Creation of successive mesh resolution levels by successively subdividing an initial triangulated torus.

recursively built, starting at the finest resolution level. The typical scaling function for this initial level consists of a delta function equal to 1 at its center and 0 everywhere else. Then, a scaling function from resolution j^{th} is computed as a linear combination of scaling functions defined on the $j+1^{th}$ resolution mesh. These functions, $\varphi_{j,k}$, are typically "hat-shaped" and their amplitude varies from 1 at their center to 0 at the vertices from the same resolution level as the center vertex k and that share an edge with vertex k.

Wavelet basis functions will be used to encode the signal differential between two successive resolution meshes. At resolution level j, a wavelet function $\psi_{j,m}$ is defined for every new node (or vertex) $m \in M(j)$. The value at a vertex v of a wavelet function from the resolution level j is computed as a combination of scaling functions from levels j and $j+1$:

$$\psi_{j,m}(v) = \varphi_{j+1,m}(v) - \sum_{k \in Neigh(j,m)} s_{j,k,m}\varphi_{j,k}(v) \tag{2.1}$$

where $Neigh(j,m)$ is a neighborhood of node m. The structure of this neighborhood defines the type of *dual lifting* [87] of the wavelet decomposition. In this work, this neighborhood will be composed of eight nodes from coarser levels (i, $i < j$). This model is referred to as the *butterfly scheme* in [87] and is presented in Figure 2.5. It

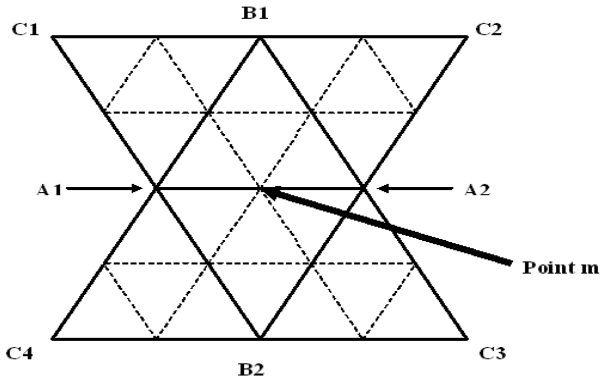
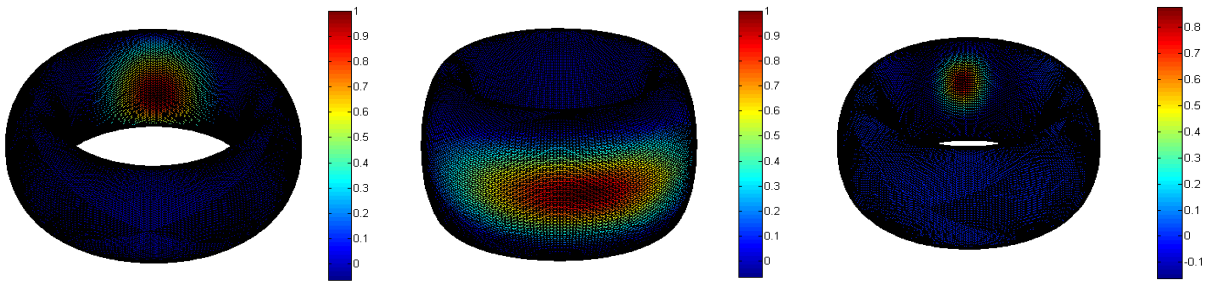


Figure 2.5: "Butterfly" neighboring system for a point m at a level j: its neighbors A1, A2, B1, B2, C1, C2, C3 and C4 are vertices from coarser levels and are used to compute its wavelet coefficient value and basis function. Dashed lines form the new triangles of level j.

has been preferred to other schemes because of its tendency to increase the smoothness of the basis functions. The two direct parents A_1 and A_2 are the endpoints of the edge to which m belongs. The two other vertices that form the two adjacent triangles with A_1 and A_2 are B_1 and B_2. Finally, the remaining parents from level $j-2$, C_1, C_2, C_3 and C_4 complete the neighborhood.

In Equation (2.1), the coefficients $s_{j,k,m}$ translate the application of a *lifting scheme*. For spherical wavelets [87], this scheme brings significant enhancement to the wavelet representation. Their definition insures that the wavelets bases have one vanishing moment.

These wavelet basis functions are built on a torus equipped with a multi-resolution mesh (as described above). Visualization of several wavelet basis functions is shown in Figure 2.6. One can observe how the support of the basis function decreases as the resolution increases.



(a) (b) (c)

Figure 2.6: (a): Scaling function, position 3 (coarser level), (b): Scaling function, position 26, (c): Wavelet function, resolution 1, position 16.

2.2.2 Surface Encoding

The set of functions composed of the scaling functions $\varphi_{0,k}$ defined on the coarsest resolution mesh and all wavelet functions $\{\gamma_{j,m}\}$ form a basis for the space of all functions of finite energy defined on a mesh. Thus, any scalar function F defined on a multi-resolution mesh can be decomposed as a linear combination of basis functions and coefficients [87]:

$$F(v) = \sum_{k} \lambda_{0,k}\varphi_{0,k}(v) + \sum_{0\leq j}\sum_{m} \gamma_{j,m}\psi_{j,m}(v) \tag{2.2}$$

The corresponding coefficients $\lambda_{0,k}$ and $\gamma_{j,m}$ encode the signal at the various resolution levels. The coefficients $\gamma_{j,m}$ are calculated by inner product between the data function F and a *dual* wavelet function $\psi^{-1}_{j,m}$, that is defined such that $< \psi_{j,m}, \psi^{-1}_{l,n} >= 1$ when $j = l$ and $m = n$ and $< \psi_{j,m}, \psi^{-1}_{l,n} >= 0$ otherwise. The filtering process that project the original signal onto the set of basis functions is referred to as the *forward transform* and the reverse operation is called *backward transform.*

In practice, a Fast Discrete Wavelet Transform can be used to implement these two transforms. The scheme used to compute those functions and coefficients defines the type of wavelet transform. This fast transform starts at the finest level, where the λ coefficients are taken equal to the function values at the corresponding vertices. Then, going down the resolution levels, wavelet coefficients are computed using linear combinations of neighboring coefficients from the upper level of resolution. Different

options exist for the choice of the neighborhood that is considered in this computation, and these have been presented in [87]. This choice may affect the level of compression and the smoothness of the basis functions, and defines the type of *dual lifting* [87] that is applied to the wavelet scheme.

Let us now give a quick sketch of Fast Discrete Wavelet Transform:

- **Forward Transform:** This algorithm computes the λ and γ coefficients from the surface coordinates. The process starts at the finest level of subdivision, where the λ coefficients are taken equal to the vertex coordinates. The dual lifting scheme is first used:

$$\forall k \in K(j): \quad \lambda_{j,k} = \lambda_{j+1,k} \tag{2.3}$$

$$\forall m \in M(j): \quad \gamma_{j,m} = \lambda_{j+1,m} - \sum_{k \in Neigh(m)} \tilde{s}_{j,k,m} \lambda_{j,k} \tag{2.4}$$

  Next, the lifting scheme adjusts the values of the coefficients as follows:

$$\forall m \in M(j): \quad \lambda_{j,A_1} = \lambda_{j,A_1} + s_{j,A_1,m} \gamma_{j,m} \tag{2.5}$$

$$\forall m \in M(j): \quad \lambda_{j,A_2} = \lambda_{j,A_2} + s_{j,A_2,m} \gamma_{j,m} \tag{2.6}$$

  where, for a vertex $m \in M(j)$, A_1 and A_2 are the two endpoints of the parent edge of which m is the middle point. The weights $s_{j,k,m}$ are computed such that the resulting wavelet has a vanishing integral. Remarks on the potential effects of these weights on the wavelet representation are made in Section 2.4.

- **Backward Transform:** The backward process start at the coarsest level and re-builds the signal coordinates step by step, up to the finest level. During this backward process, lifting scheme is first used :

$$\forall m \in M(j): \quad \lambda_{j,A_1} = \lambda_{j,A_1} - s_{j,A_1,m} \gamma_{j,m} \tag{2.7}$$

$$\forall m \in M(j): \quad \lambda_{j,A_2} = \lambda_{j,A_2} - s_{j,A_2,m} \gamma_{j,m} \tag{2.8}$$

Next, dual lifting adjusts the values of the coefficients as follows:

$$\forall k \in K(j): \quad \lambda_{j,k+1} = \lambda_{j,k} \tag{2.9}$$

$$\forall m \in M(j): \quad \lambda_{j+1,m} = \gamma_{j,m} + \sum_{k \in Neigh(m)} \tilde{s}_{j,k,m} \lambda_{j,k} \tag{2.10}$$

where, for a vertex $m \in M(j)$, A_1 and A_2 are the two endpoints of the parent edge of which m is the middle point.

This method is fast, and efficient. The lifting scheme is easily incorporated into this algorithm. The update equations for the forward and backward wavelet transforms are similar to those proposed in [87]. Experiments show that all lifted wavelets provide more accurate compression than simple basis functions and that the dual lifting used in this work increase smoothness in the wavelet functions. However, particular attention needs to be given to the implementation of both lifting and dual lifting schemes. This point is discussed in Section 2.4.3 as low pass filtering techniques are presented.

At this point of the chapter, it has been explained to the reader how genus-one surfaces will be encoded by projecting their coordinates on a set of wavelet basis functions defined on a multi-resolution mesh. However, meshes encountered in practice rarely exhibit an appropriate subdivision connectivity. Therefore, it is very often necessary to first *re-mesh* the surfaces in order to equip them with a multi-resolution grid. The next section gives a brief overview on potential techniques that may be used to re-triangulate surfaces and then describes the method used in the proposed work.

2.3 Surface Re-triangulation

Surface re-meshing may be approached in many different ways. The existing re-meshing techniques are now briefly described.

2.3.1 Previous Work

Surface re-meshing aims to equip a surface with a new mesh, where the coordinates of the new vertices are obtained via interpolation. In order to perform this interpolation, one first needs to establish some sort of correspondence between the positioning of original vertices and the new mesh vertices. This *vertex correspondence* is key in any re-meshing algorithm and it becomes more complex as the genus of the surface increases.

Vertex correspondence may be obtained through surface morphing. Morphing (or metamorphosis) is the process of gradually changing a source object into a target object. Work on 3D mesh morphing has brought very interesting methods on ways to map a source mesh to a target mesh. Different approaches to the vertex correspondence problem exist and these are summarized in [57]. This correspondence process often necessitates the creation of a single mesh with two instantiations, one for the source and for the target surface. While some methods are topology-specific [52, 56], some others allow one to handle shapes with arbitrary genus [51, 58, 86]. The results obtained with these methods usually require that the user define features in the source mesh that will be manually mapped to the target. These features usually correspond to key points of the shape that need to be carefully mapped to the target mesh. Also, the computational complexity of these techniques often is prohibitive as multiple mesh manipulations are sometimes necessary. Finally, these methods are related to and depend on the triangulation of the target.

Another common approach to surface re-meshing consists of parameterizing the surface. Surface parametrization is the process of mapping a surface onto a plane region. Once this mapping has been established, the creation of a new mesh on the flattened region makes the interpolation with the original vertices trivial.

Surface parametrization algorithms are very commonly used in fields like texture mapping, surface flattening and re-meshing. Several different categories of surface

parametrization exist. Classical methods aim to minimize a cost function, while trying to keep low distortion [28, 46, 64, 65, 68, 83]. Distortion stems from the lack of *conformality* in the flattening algorithm. In surface mapping, conformality is defined as the preservation of angles on the map. Several methods have been developed that aim to respect this property. These are refer to as conformal (or quasi-conformal) parameterizations. In practice, the conformal parametrization of a given manifold is usually approximated as surfaces are represented by triangulated meshes. For this type of algorithm, one needs to make sure that conformal mapping remains intrinsic to geometry and independent of triangulation and resolution. Haker *et al.* proposed using conformal mapping to parameterize spherical surfaces [39] and surfaces equivalent to open-ended cylinders [40]. The extension of conformal parametrization to higher genus surfaces is much more complex. Gu *et al.* proposed an algorithm to compute conformal structures of nonzero genus closed surfaces [35,36]. In [35], theory on a possible approximation of the De Rham cohomology and the computation of holomorphic one-forms is given. Details on the geometric realization are provided in [36].

Area preserving mapping methods tend to preserve the relative areas of the different regions of the surface (and therefore triangle areas for triangulated meshes) in the flattening map. When flattening constitutes the first step of a re-meshing process, it may be important to observe that areas are relatively well preserved. Indeed, if very large areas of the original surface are mapped to a very small portion of the resulting flattened surface, vertex interpolation may be cumbersome in these regions. It is mathematically impossible to obtain perfect conformality as well as area preservation in the same surface mapping algorithm. However, one may think about building a flattening map that finds an optimal balance between area-preservation and conformality. Using calculus of variation tools, area preserving diffeomorphisms can be found that minimize distortion [6]. One may also think about adjusting the

flattening methodology by applying an area-correcting algorithm to the parametrization obtained with conformal mapping. Zhu *et al.* [103] proposed using optimal mass transport (O.M.T.) theory for the flattening of open-ended cylinder-like surfaces.

In the proposed work, conformal mapping and area-correction are used in order to parameterize genus-one surfaces that a cut has made topologically equivalent to a planar rectangle (i.e. an unfolded torus). Parametrization follows the framework defined in [40] and [103]. Once the mapping is completed, interpolation is used to replace the existing mesh with a multi-resolution grid. This type of mesh connectivity offers the opportunity to develop a wavelet scheme that will decompose the signal in space and scale.

2.3.2 Proposed Methodology

The input of the proposed algorithm is a 3D triangulated genus-one surface. As explained in Section 2.2.1, the wavelet decomposition of the surface requires a multi-resolution mesh structure. The algorithm used in this work to re-triangulate the surface follows the following steps:

- Cut the surface along a homology basis: The cut surface becomes topologically equivalent to an open-ended cylinder.
- Find conformal map between the surface and a rectangular plane target: Conformal mapping is implemented using the method proposed by Haker *et al.* [40].
- Adjust the areas of the flattened triangles: This operation tends to correct the area distortion between the original surface and the result of the flattening algorithm. The method used here is based on the work of Zhu *et al.* [103].
- Equip the planar rectangle with a regular multi-resolution mesh by successively subdividing an initial coarse mesh: The new mesh structure will allow for wavelet decomposition.

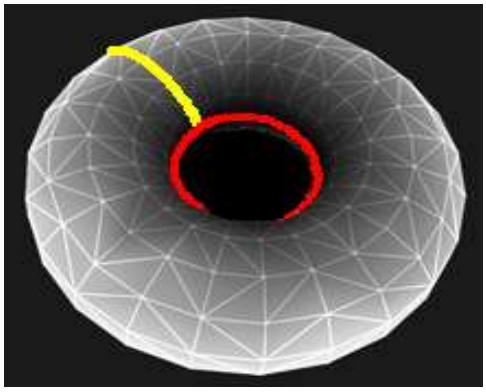


Figure 2.7: Homology basis on a genus-one torus. The red and yellow lines represent two loops, representing the two homologous classes

- Re-mesh the original surface with the new regular mesh using vertex interpolation: After the re-meshing process, the new vertices on the surface will exhibit a multi-scale connectivity.

2.3.2.1 Cut in the Surface

One way to find an appropriate cut on the triangulated surface is to refer to the homology group of the surface. A homology basis characterizes the topology of the surface and can be denoted by a set of loops $e_1, e_2, e_3, ..., e_{2g}$, where g is the genus of the surface. Each loop of this set is representative for the homologous class to which it belongs to. Intuitively, two loops on a surface are homologous if one can be deformed into the other while always keeping it entirely on the surface. Figure 2.7 shows the homology basis for a genus-two torus.

For low genus surfaces, finding such a cut is relatively intuitive and this can be manually executed by the user. Otherwise, methods exist that automatically search for a homology basis [72]. In this case, Dijkstra's algorithm may be used to shorten each base loop.

After this cut has been made, the surface is equivalent to an open-ended cylinder. In [40], a method for mapping this type of surface to a flat rectangle in a conformal

manner is presented. The next section briefly describes this technique.

2.3.2.2 Conformal Mapping: Background and Implementation

Background and Theory Conformal mapping is a very commonly used concept in complex analysis. It is used in many different fields, such as physics or engineering. A conformal mapping consists of a transformation f in the complex domain. $f : \Sigma \rightarrow \mathbf{C}$ sends the surface Σ to the complex plane. The real and imaginary parts of the mapping, respectively $u = u(x, y)$ and $v = v(x, y)$, must be two *conjugate harmonic functions*. This implies that:

- The first and second order partial derivatives $\frac{\partial u}{\partial x}$, $\frac{\partial v}{\partial y}$, $\frac{\partial^2 u}{\partial x^2}$, and $\frac{\partial^2 v}{\partial y^2}$ exist and be continuous
- Both functions satisfy the *Laplace equation*:

$$\Delta u = 0 \tag{2.11}$$

$$\Delta v = 0 \tag{2.12}$$

- The *Cauchy-Riemann equations* be satisfied:

$$\frac{\partial u}{\partial x} = \frac{\partial v}{\partial y} \tag{2.13}$$

$$\frac{\partial u}{\partial y} = -\frac{\partial v}{\partial x} \tag{2.14}$$

Implementation In the problem of mapping the tubular structure to a rectangle, boundaries need to be paid particular attention. The boundary of the surface Σ is composed of two circles σ_0 and σ_1. The implementation of this conformal mapping is based on the work proposed by Haker *et al.* [40], which uses finite elements to find the flattening map f for the triangulated surface.

The first step of the mapping algorithm will consist in finding a solution to the equation $\Delta u = 0$, with the following boundary conditions: $u = 0$ on σ_0 and $u = 1$

on σ_1. It is known that the solution of the Laplace equation $\Delta u = 0$ is a harmonic function that minimizes the following Dirichlet functional:

$$D(u) = \frac{1}{2} \int \int_{\Sigma} |\nabla u|^2 dS \tag{2.15}$$

$$u|_{\partial\sigma_0} = 0, u|_{\partial\sigma_1} = 1 \tag{2.16}$$

In order to solve this minimization problem on the triangulated surface, u will be approximated by a piecewise linear function that is continuous on Σ and linear on each triangle. If V defines the index of an arbitrary vertex of the triangulated surface, u may be written as follows:

$$u = \sum_{V} u_V \phi_V \tag{2.17}$$

where ϕ_V is a basis function, from the basis ϕ_V, which is defined as follows:

$$\begin{array}{ll} \phi_V(V) & = 1 \\ \phi_V(W) & = 0 \text{ for } W \neq V \\ \phi_V \text{ is linear on each triangle} & \end{array}$$

Once a solution u has been found to the aforementioned Dirichlet problem, a curve C, running from σ_0 to σ_1, is defined such that u is strictly increasing along C. A cut is then performed along this cut. The surface $\Sigma \backslash C$ therefore becomes equivalent to a planar rectangle.

Next, a similar problem is solved for v. Assuming that u and v are conjugate harmonic functions, the Cauchy-Riemann equations ((2.14)) are used in roder to compute boundary values for v by integrating $\frac{\partial u}{\partial x}$ along the newly created closed boundary. Thus, v can now be found by solving the same Dirichlet problem as for u.

The mapping $f = u + iv$ finally send the surface Σ to a rectangle.

2.3.2.3 Area Correction using Optimal Mass Transport

The conformal mapping algorithm described in Section 2.3.2.2 preserves angles and local geometry. However, areas are not preserved and this becomes a serious inconvenience for further re-meshing of the surface. Indeed, some regions have been

significantly shrunk while others enlarged. Therefore, when re-meshing the domain with a regular grid by interpolation, it may be very hard to "capture" the surface details contained in the shrunk regions, while other regions will be over-sampled. It is important to note that area distortions become worse for complex surfaces that exhibit high curvature regions. In this section, a methodology for (partially) correcting the area distortion introduced by the conformal mapping is presented.

In order to quantitatively assess this area distortion, this work focuses on the ratio of the area of a triangle on the original surface over the area of the same triangle on the flattened surface (referred to as *area-changing ratio*). This quantity may be defined as a *mass density*. This density, obviously uniform on the original surface Σ, was then deformed by the conformal mapping f, while the total mass was preserved. Let us denote this density as μ_0 on the flattened surface. Obviously, the value of μ_0 will be greater than one in the regions that were enlarged by the mapping, and smaller than one in the regions that were shrunk. The objective would be to find a one-to-one mapping m that would transform this distribution into a much more uniform density μ_1, while conserving the total *mass* (i.e. the total area of the triangulated surface). The composition of f and m will result in a one-to-one area-preserving mapping: $g = m \circ f$.

Zhu *et al.* [103] proposed an interesting approach to this problem. In their work, Optimal Mass Transport (O.M.T.) is used in order to obtain a mass preserving mapping to correct the area distortion introduced by conformal flattening. The next paragraphs aim to give a brief description of that method.

Background and Theory

First, μ_0 is interpolated on a rectangular grid that covers a rectangular region Ω_0, while the target distribution μ_1 is defined on a similar rectangular region Ω_1. The

mass preservation principle can be characterized by the following equality:

$$\int_{\Omega_0} \mu_0 = \int_{\Omega_1} \mu_1 \tag{2.18}$$

In the present work, we would like to find a *mass preserving* mapping m between (Ω_0, μ_0) and (Ω_1, μ_1). Mass preserving mappings are diffeomorphisms m that verify the following property:

$$\mu_0 = |Dm| \mu_1 \circ m \tag{2.19}$$

A mapping that satisfies this property will lead to a redistribution of a mass of material from one distribution (Ω_0, μ_0) to another (Ω_1, μ_1). This formulation is an extension of the very well-known Monge-Kantorovich problem. This problem was about finding the optimal way, in a sense of minimal transportation cost, of moving a pile of soil from one place to another. Thus, in this work, the objective will be to find the optimal mapping $\tilde{m}$ among all possible mass-preserving mappings m. Though, optimality is only defined with respect to a metric that needs to be chosen by the user. This metric will define the type of penalty that will be placed on each bit of material that is moved by the map m. Thus, the cost incurred for "transporting the mass" is minimized. As suggested in [103], this work will use the L^2 Kantorovich-Wassertein metric, which is defined as follows:

$$d(\mu_0, \mu_1) = \liminf_{m \in MP} \int \| m(x) - x \|^2 \mu_0(x) dx \tag{2.20}$$

A theoretical result shows that there exists a unique optimal mass preserving mapping $\tilde{m}$ that solves this L^2 minimization problem [11, 33, 55]. This solution can be written as the gradient of a convex function ω, which will therefore verify the Monge-Ampère equation: $|H\omega| \mu_1 \circ (\nabla \omega) = \mu_0$. The methodology used to find the optimal solution $\tilde{m}$ exploits results on *polar factorization*. If m^0 denotes an initial mass preserving mapping, it has been shown that this mapping can be written as

follows:

$$m^0 = (\nabla\omega) \circ s \tag{2.21}$$

where s is also a mass preserving mapping. The following algorithm is iterative. Starting with m^0, the mapping s will vary in order to form another mass preserving mapping $m = m^0 \circ s^{-1}$. When considered as a vector field, m can be decomposed as the sum of a curl-free and a divergence-free field: $m = \nabla\omega + \chi$. If one can find $s = \tilde{s}$ such that χ goes to zero, then the resulting m will end up begin a curl-free mass preserving mapping ($m = \nabla\omega$). By uniqueness of the Monge-Ampère equation solution, this $s = \tilde{s}$ would lead to the optimal solution $\tilde{m} = \nabla\omega = m^0 \circ \tilde{s}^{-1}$ (this is equivalent to finding the optimal polar factorization of m^0). Thus, by rearranging the initial solution by composing it with a mass preserving mapping, the optimal solution can be iteratively be found.

A technique to find an initial mapping m^0 is given in [103]. Then, after derivations, it can be shown that the evolution equation for m, that will converge to the optimal solution with respect to the L^2 metric, can be written as follows:

$$m_t = \frac{2}{\mu_0} Dm \nabla^{\perp} \triangle^{-1} div((m - id)^{\top}) \tag{2.22}$$

where id is the identity map, $\perp$ a rotation by $\pi/2$ counter-clockwise and t refers to time and is used as a subscript to denote differentiation with respect to time.

Implementation For these surfaces, the aforementioned mass preserving algorithm is applied to the flattened triangulated region that resulted from applying conformal mapping to the original surface. μ_0 is defined as the ratio of the area of a triangle on the original surface over the area of the same triangle on the flattened surface. The domain of definition Ω_0 for the mass preserving mapping will consist of the planar rectangular area obtained after conformal flattening. The optimal mass transport algorithm will be implemented on a regular rectangular grid. Therefore,

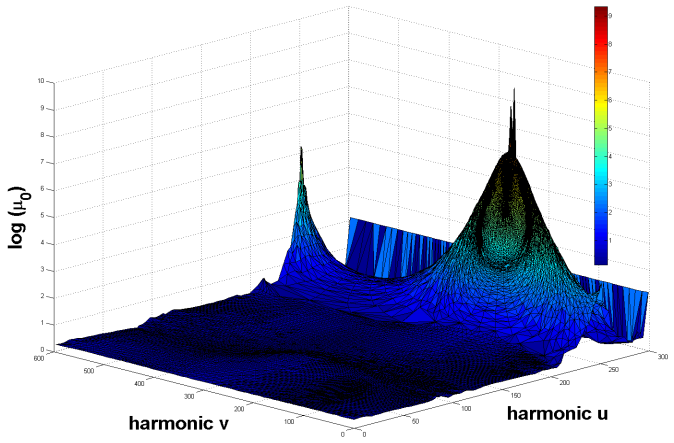
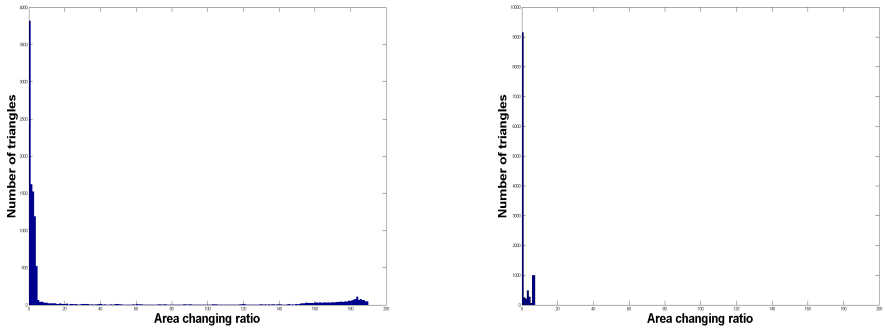


Figure 2.8: Distribution of μ_0 for the TEAPOT shape.

μ_0, defined for each flattened triangle, is computed at every vertex of the rectangular grid by interpolating its values from the triangulated mesh.

For surfaces such as those that have been analyzed in the present work, μ_0 exhibits prohibitive distortions. Figure 2.8 shows the distribution of μ_0 for the TEAPOT shape. The large variations of μ_0 observed in this example stems from the high curvature that some parts of the original shape exhibits. These high variations in the distribution of μ_0, as well as the sparsity of this distribution make the implementation quite challenging. More particularly, the resolution of the rectangular grid needs to be defined so that it appropriately captures these variations. If this condition is not met, the uniform target μ_1 will become impossible to approach. Also, significant deformations of some rectangles from the grid may introduce problems of overlapping triangles.

The algorithm is applied to (Ω_0, μ_0). Because of the discretization of the problem for implementation purposes, convergence to a completely curl-free mapping is not possible in practice. In order to achieve a decent final mapping, a criterion is set up so that the iterative process stops as soon as the curl term drops under a certain level.



(a) (b)

Figure 2.9: Distribution of the area changing ratios. (a): before mass transport mapping and (b) after mass transport mapping

Changes in the triangle spatial distribution on Ω_0 can be observed by a reverse bi-linear interpolation from the deformed rectangular grid. In Figure 2.9, two histograms show the distribution of the area-changing ratios before and after area correction. It appears that the area of the triangles has been considerably corrected (more triangles have a ratio close to 1). However, this correction is not perfect in practice. First, numerical errors are introduced by the re-sampling with the rectangular grid. Second, initial prohibitive area distortions make such a result very difficult to even approach.

2.3.2.4 Re-meshing

In this section, the description of the actual re-meshing process is described in details. The re-meshing process aims to equip surfaces with a multi-resolution mesh that has been built through N successive subdivisions of an initial triangulated mesh as described in Section 2.2.1. The resulting mesh is said to have a *multi-scale connectivity*. In this type of grid, all the nodes are organized by level of subdivision as each vertex belongs to one of the $N + 1$ resolution levels. Once a surface is equipped with a multi-scale connectivity mesh, the construction of a set of wavelet functions and the encoding of any signal defined on the surface are made feasible (as explained in Section 2.2.1).

At this stage of the method, the surface has been flattened and mapped to a

planar rectangle. By equipping the same rectangular plane with a new mesh, one may easily establish a correspondence map between the positioning of the flattened surface vertices and those of a new mesh structure. Thus, a simple bilinear interpolation algorithm allows the spatial coordinates of the original surface to be calculated at these new vertices. The new mesh is built by successively subdividing an initial triangulated grid.

Now, particular attention needs to be given to the boundary conditions of this re-meshing process. Indeed, the original surface is topologically equivalent to a torus and not to a planar rectangle. The new mesh is built such that, when folded to form a torus, boundary vertices coincide and periodicity is satisfied. Thus, it is important that the boundary vertices of the flattened surface also satisfy the periodicity conditions. More precisely, the boundaries correspond to the two cuts that have been made on the surface when building the flattening map and that have been preserved by the mass-preserving mapping. The first cut was made along a homology basis of the surface and the corresponding pair of boundaries is (σ_0, σ_1). These two boundaries do not actually contain the same set of vertices. Indeed, the triangles that have been removed to facilitate the conformal flattening, connect these two boundaries. Therefore, it is necessary to add these back to the flattened mesh. The second cut was running from σ_0 to σ_1, following increasing values of u (the real part of the mapping f). The two boundaries of the second cut are composed of the same set of vertices. In order to satisfy the periodicity conditions, the vertices of the two boundaries must match. More particularly, if a and b respectively denote the real and imaginary parts of the mapping g, the two pairs of boundaries may be represented as $[(A_0 : a = 0), (A_1 : a = max(a))]$ and $[(B_0 : b = 0), (B_1 : b = max(b))]$. With these notations, the periodicity conditions become: $b(A_0) = b(A_1)$ for the vertices of the first cut and $a(B_0) = a(B_1)$ for the vertices of the second cut. If these conditions are not met, then adjustments need to be made to the positions of these vertices.

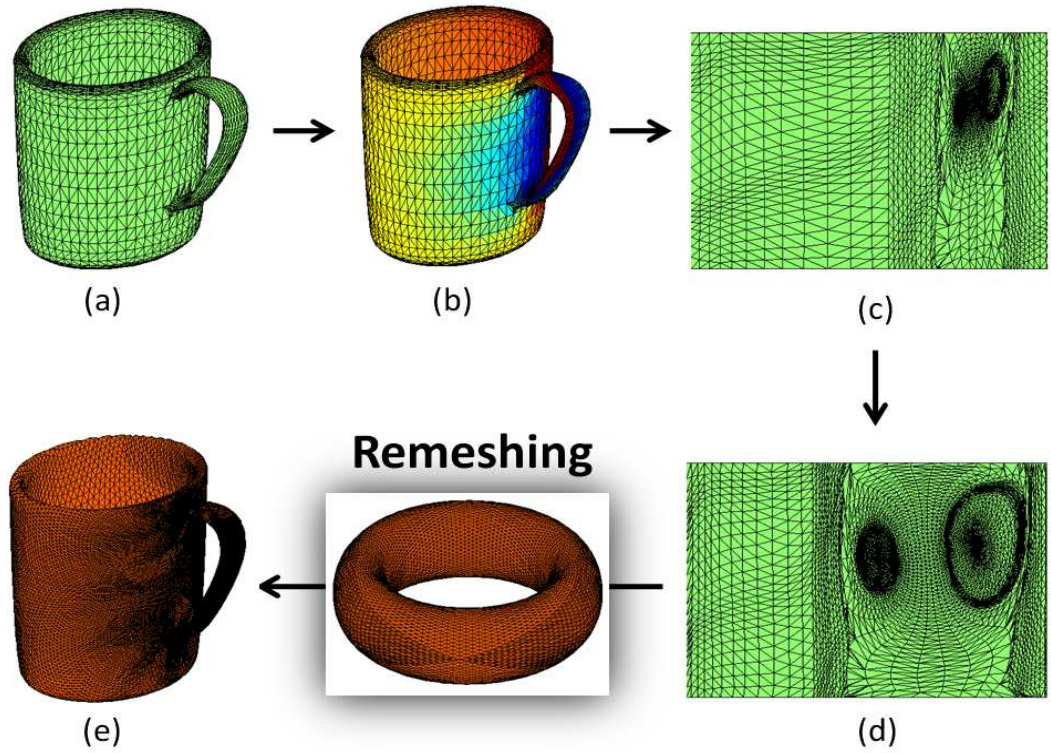


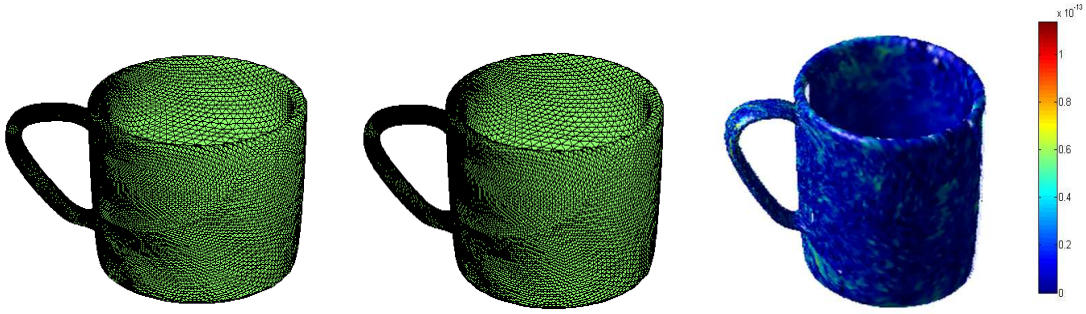
Figure 2.10: The different steps of the re-meshing process: (a) The surface is represented by a 3 mesh. A cut can me be made along a homology basis to make it equivalent to a tubular structure, (b) Conjugate harmonic functions are computed by solving two successive Dirichlet problems. Here, the color represents the v harmonic function on the surface, (c) The surface is flattened using conformal mapping, (d) Optimal Mass Transport adjusts the areas of the triangles, and (e) By interpolation, the surface is re-triangulated with a regular multi-resolution mesh

If the boundary conditions are met, re-meshing can be conducted by interpolation. Through this process, the vertices of the new mesh will be assigned surface coordinates that are interpolated values of the original mesh coordinates.

Figure 2.10 summarizes the different steps of the pre-processing and re-meshing process.

2.4 Experiments: Wavelet Encoding of Genus-One Surfaces

In this section, results are shown for two different shapes: the TEACUP and the TEAPOT shapes. Both surfaces have been re-meshed with a grid composed of 32768



(a) (b) (c)

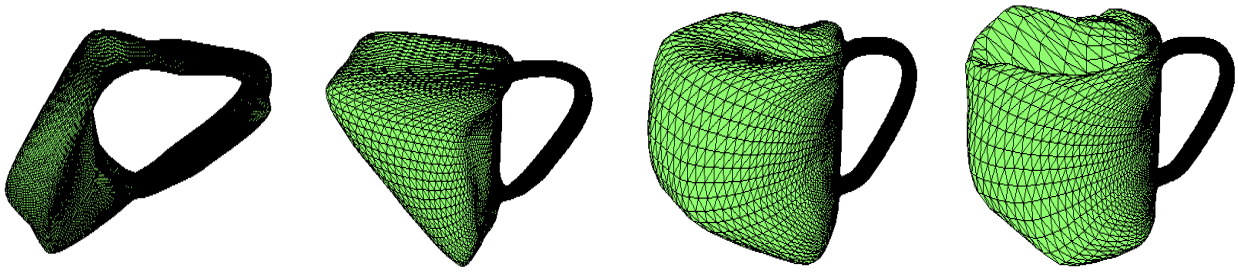
Figure 2.11: Reconstruction of the surface after forward and backward wavelet filtering. (c) displays the reconstruction error (dark blue corresponds to the lowest error and red to the maximal values).

vertices. This mesh has been obtained after 5 successive subdivisions of an initial triangulated grid composed of 32 nodes.

In order to show how this wavelet filter works on a triangulated surface with genus one, the forward wavelet transform has first been applied. Next, the inverse wavelet transform was used to rebuild the surface. The efficiency of the filter may be verified by comparing the reconstructed surface with the original one. This is shown in Figure 2.11. Figure 2.11-(b) displays the shape of the surface after the forward and backward wavelet filters have been applied to the input surface (a). In Figure 2.11-(c), the point-to-point reconstruction error is represented on the surface, where dark blue corresponds to the minimal error (in % of the bounding box) and red to the maximal values. The very low level of error confirms that the wavelet encoding scheme can be considered as a lossless process.

2.4.1 Low-Pass Filtering

Even though, this decomposition is already useful in itself by allowing one to represent the surface at various resolution levels, flexibility in the level of detail can be reached by projecting the surface coordinates into a limited set of basis functions. Concretely, if higher levels of resolution, i.e., higher frequency variations, are aimed to be disregarded by the user, coefficients from the corresponding levels may be "zeroed". This



(a) (b) (c) (d)

Figure 2.12: Various levels of low-pass filtering applied to the TEACUP shape. (a)-to-(d) represent successive low pass filters applied at increasing resolution levels. (a) displays the coarser version of the shape and (d) the finer.

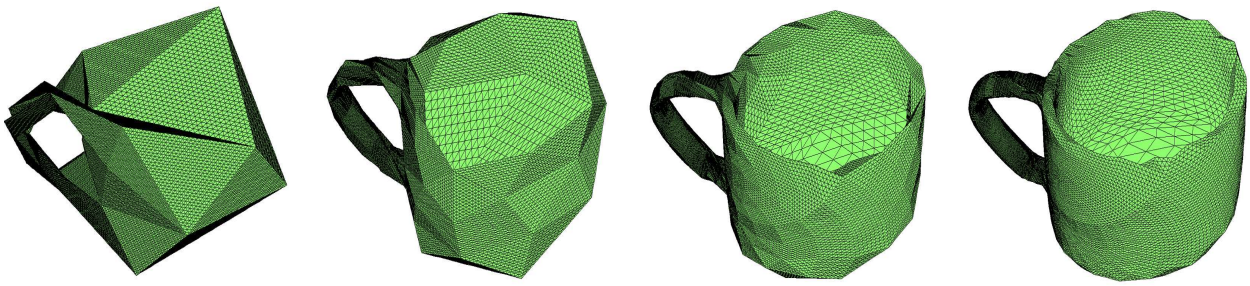
procedure is equivalent to annihilate the basis functions from higher resolution levels. Namely, this can be seen as a low pass filter of the signal. Figure 2.12 shows the effect of low-pass filters on the shape of the reconstructed surface for the TEACUP example. This type of filter is applied at multiple levels. Figure 2.12-(a) only shows the low frequency variations of the shape by projecting the signal on the set of scaling functions only. Figures 2.12-(b)-to-(e) successively add a finer level of representation as basis functions of higher resolution are added to the projection set one-by-one.

Due to the local support of the wavelet basis functions, this filtering concept may be considerably refined if the algorithm were suppressing coefficients in an individual manner instead of removing all coefficients from a given level. This type of local filtering may be obtained by applying *wavelet shrinkage* to the set of wavelet coefficients. In general, wavelet shrinkage consists of removing the noisy part of a wavelet coefficient. In the context of filtering and compression, one may simply want to either keep or completely eliminate a given coefficient. This is referred to as *hard shrinkage*. Thus, only a reduced set of coefficients would be used to encode the original signal in the wavelet domain. Obviously, in this type of procedure, a trade-off needs to be found between the rate of compression and the accuracy of the remaining signal. Several different shrinkage approaches exist. A novel wavelet shrinkage model for shape de-noising is proposed in this thesis and fully described in Chapter 3.

In the case of genus-one surfaces, such as the TEACUP or TEAPOT shapes presented in this chapter, the area-distortion introduced by the conformal-mapping that precedes the re-meshing process is sometimes significantly high. Due to the difficulties one may have to find an appropriate rectangular grid for the mass transport algorithm, a good area correction becomes more challenging to reach and the distribution of the vertices on the resulting mesh does not necessarily meet the theoretical area-preserving requirements. Thus, the re-meshing phase needs to be performed on this imperfect flattened surface and one may need to increase the resolution of the new mesh in order to capture all relevant features of the surface during the re-triangulation process. Therefore, some regions of the surface will turn out to be over-sampled while others will have just enough points. When the signal is then encoded in the wavelet domain, most of the fine resolution coefficients from the over-sampled regions will be insignificant and will be subject to be shrunk. Thus, applying individual low pass filters, or wavelet shrinkage, to the set of signal encoding coefficients should considerably help compress the signal without altering the shape of the surface.

2.4.2 Remarks on the Dual Lifting Scheme

It was mentioned in Section 2.2.2 that wavelet coefficients were computed using linear combinations of neighboring coefficients from the next lower level of resolution and that various options existed for the choice of the neighborhood. This choice may affect the level of compression and the smoothness of the basis functions. It defines the type of *dual lifting* [87] that is applied to the wavelet scheme. The dual lifting chosen in the proposed work tends to increase the smoothness of the wavelets functions. This particularity can be observed by comparing the proposed wavelet decomposition with that of the same surface using another dual lifting scheme. For this purpose, let us consider the *linear* dual lifting model that attributes different values to the weights $\tilde{s}_{j,k,m}$: $\tilde{s}_{j,k,m} = 1/2$ at the two parent sites ν_1 and ν_2 of the coefficient $m \in M(j)$



(a) (b) (c) (d)

Figure 2.13: Various levels of low-pass filtering applied to a linear dual lifting schemed wavelet representation of the TEACUP shape. (a)-to-(d) represent successive low pass filters applied at increasing resolution levels. (a) displays the coarser version of the shape and (d) the finer.

and $\tilde{s}_{j,k,m} = 0$ otherwise. The lack of smoothness of the wavelet functions in this alternative model can be observed by applying the same low-pass filtering process as that presented in paragraph 2.4.1. The results of this filtering at different resolution levels are shown in Figure 2.13.

The smoothness of the wavelet basis functions is a desirable property. However, as mentioned in the next paragraph (Section 2.4.3), broader neighborhoods, such as that of the *butterfly* dual lifting scheme, may lead to undesirable outcomes in certain cases (e.g. non-uniformity of the mesh resolution over the entire surface). Therefore, a model such as the linear dual lifting may be sometimes considered.

2.4.3 Remarks on Low-Pass Filtering and Lifting Schemes

It is important to note here that the application of a low pass filter in the wavelet domain may sometimes lead to undesirable results. Let us first explain this issue in detail, and to do so, let us assume that a given wavelet coefficient $\gamma_{j,m}$ has been shrunk to zero within the filtering process. First, the lifting scheme that had been applied to λ_{j,ν_1} and λ_{j,ν_2} (Equations (2.5) and (2.6)) during the forward wavelet transform is not anymore "compensated" for in the first phase of the reverse transform (Equations (2.7) and (2.8)). This difference in the values of λ_{j,ν_1} and λ_{j,ν_2} is proportional to the amplitude of the weights $s_{j,\nu_1,m}$ and $s_{j,\nu_2,m}$. Next, since the same observation

can probably be made for other neighbors of j, the sum $\sum_{k \in Neigh(m)} \tilde{s}_{j,k,m} \lambda_{j,k}$ may have a value very different from what it would have been without low pass filtering. The "filtered" value of $\lambda_{j+1,m}$ may therefore be considerably affected. Finally, the amplitude of the original coefficient $\gamma_{j,m}$, now reduced to zero, will also have a relevant effect on $\lambda_{j+1,m}$ (Equation (2.10)). To sum up all these comments, one should say that low pass filtering may considerably affect the reconstructed signal and that the intensity of these alterations may be explained by the following factors: the original intensity of the wavelet coefficient, the amplitude of $\lambda_{j,k}$ at the neighboring sites (dual lifting scheme) and the values of the lifting weights $s_{j,k,m}$ (lifting scheme). These factors may vary with the structure of the mesh, as well as with its level of refinement. Let us now focus on the potential impact of such a low pass filter, that would be applied to the set of wavelet coefficients and shrink the value of $\gamma_{j,m}$ to zero. For this, let us consider the TEACUP shape and bottom part of the "bowl" part of the shape, which is composed of an interior S_i and an exterior S_e face. These plane regions are parallel and very close to each other in space. Let us assume that $\gamma_{j,m}$ is defined at a vertex $m \in S_i$. If, for some reason, the part of the mesh around m is under-sampled, it may happen that the low pass filter "pushes" the spatial coordinates of m towards S_e, such that the reconstructed S_i and S_e undesirably intersect. Thus, to conclude this discussion, it is important to say that the implementation of both lifting and dual lifting schemes requires particular attention. Particularly, a regular and fine triangulation, adapted to the complexity of the surface, is usually recommended in order to avoid undesirable outcomes.

2.5 Concluding Remarks on Non-Spherical Shape Analysis

In this chapter, it was shown that wavelet analysis using second generation wavelets can be applied to signals defined on genus-one triangulated surfaces.

Re-meshing is a key element of this type of algorithm when the original mesh

does not exhibit a multi-resolution connectivity. The methodology described in this chapter proposes using a combination of conformal flattening with an area-preserving mapping. This algorithm appears to be *almost* fully automated manner. Indeed, only the initial cut in the surface may require supervision from the user.

For surfaces with high-curvature regions, the conformal flattening part of the parametrization often introduces significant area-distortions. Area correction becomes more challenging to perform with an unsupervised area-preserving mapping function. Thus, directions for future research may include the potential development of alternatives to the present re-meshing process.

Another direction for future research would be analyzing the significance and legitimacy of the lifting scheme in the development of second generations wavelets on non-spherical shapes.

CHAPTER III

3D SURFACE ENHANCEMENT USING WAVELET THRESHOLDING

This chapter presents a framework for smoothing and compressing 3D surfaces represented by polygonal meshes. A statistical wavelet-based methodology is proposed that decomposes the surface in a multi-scale manner and then separates noise-like elements from relevant parts of the signal.

This work finds particular applications in shape analysis for medical imaging, where acquired data (e.g. MRI) often carries irrelevant and noise-corrupted information. Many structures in the body are topologically spherical. Therefore, in this article, primary attention is given to zero-genus surfaces, i.e. surfaces with a spherical topology. However, the proposed shrinkage has also been tested on non-spherical shapes based on the wavelet decomposition presented in Chapter II.

The first section of this chapter aims to motivate and explain the objective of the proposed methodology. This first section describes the advantages of mesh approximation, briefly reviews past work that is relevant to the topic and presents the major contributions of this work. Next, two surface enhancement models will be presented in Sections 3.4 and 3.5, and experimental results will be provided in Section 3.6. Finally, several concluding remarks will be made.

3.1 Motivation for a Multi-scale Smoothing and Compression Model

In this section, the notion of mesh approximation is introduced and the need for enhancement is justified. Next, relevant work presented from the literature is briefly

reviewed. Finally, the motivation for a multi-scale mesh enhancement model is explained and a summary of the contributions of the proposed work is given.

3.1.1 Motivation for Mesh Enhancement

Meshes provide an easy way to approximate surfaces through discretization. The choice of a mesh size allows one to keep control of the smoothness and the complexity of the shape signal.

For any smooth surface represented by a polygonal mesh, the coordinates of neighboring vertices are usually inter-dependent. Therefore, it is reasonable to assume that redundant information exists within the surface data set. In addition, shapes may be affected by some undesired noise, which is characterized by high-frequency and irrelevant features on the surface. These two issues lead us to consider two types of representation improvements: mesh simplification and surface smoothing. On the one hand, mesh simplification aims to eliminate redundancy and to minimize the amount of data needed to encode the shape signal. On the other hand, surface smoothing techniques enable one to *de-noise* the shape signal by removing irrelevant high-frequency artifacts. The proposed research aims to fulfill these two objectives while keeping an accurate approximation of the shape signal. A trade-off needs to be found between signal enhancement and accuracy of the resulting encoding.

3.1.2 Prior Work on Classical Mesh Enhancement

This section briefly describes and reviews several relevant publications from the literature that address the problem of 3D mesh compression and smoothing.

Elimination of information redundancy can be done through mesh simplification. This encompasses a lot of different *filter-based* techniques, which all consist in removing the vertices that do not truly add up to the signal and in rearranging triangles accordingly. *Facet merging* methods [38, 44, 50] search for coplanar triangles and reduces to a minimum the number of triangles needed for covering the given sub-part

of the surface. *Decimation* [88] refers to iterative methods that successively consider each vertex for potential elimination based on a pre-defined decimation criteria, which depends on the type of vertex that is considered (boundary, interior, complex, simple). *Re-tiling* [97] consists of inserting new vertices into the mesh at random, moving them around in order to place more points in high-curvature areas, removing the old vertices, and finally reconnecting the remaining points to obtain a new triangulated mesh. In *global energy function optimization* methods [28,45], an energy function usually models the tradeoff between accuracy and complexity. Optimizing this energy function therefore consists of minimizing the distance between the given mesh and the ground-truth shape, while keeping low the number of vertices used in the mesh representation. Finally, *vertex clustering* groups vertices based on geometric proximity but does not necessarily preserve the topology, which can be a real issue. All these geometric methods offer very powerful ways for both simplifying mesh connectivity and compressing shape data.

Surface enhancement may also refer to signal smoothing, which can be achieved using signal processing approaches. Laplacian smoothing, one of the most common techniques for 3D mesh smoothing, is an iterative algorithm that locally moves vertices according to the value of the Laplacian operator. This operator is defined for each vertex as a function of its direct neighbors. This method is fast and simple to implement, but very often shrinks surface volumes undesirably. To prevent this drawback, several methods have been proposed, among which those described in [23,96,99]. Another well-recognized method for surface smoothing and fairing is the use of the discrete mean curvature flow. Vertices are moved along the surface normal with a speed equal to the local mean curvature [22, 23, 89].

Most of these aforementioned techniques aim to either smooth or simplify meshes but not systematically do both. On the one hand, mesh simplification methods mainly enable one to get rid of irrelevant vertices but do not necessarily smooth the surface

as desired. On the other hand, Laplacian-based and mean curvature flow methods smooth the original surface without removing vertices or compressing data. The goal of the proposed work is to combine both types of surface enhancement, namely compression and smoothing, in an efficient and automated manner.

3.1.3 Motivation for a Multi-scale Model and Background

This thesis proposes looking into multi-scale approaches for efficient surface compression and de-noising. First, it is important to note that multi-resolution mesh structures have very nice properties as their connectivity is intrinsically encoded in a straightforward and elegant manner. Indeed, this type of grid can be decomposed into a set of nested resolution levels, where nodes from a given level are connected to those of coarser scales through a very regular construction scheme (as described in Chapter II). In practice, a multi-scale mesh is often built iteratively by successively adding nodes to an initial coarse grid. In order to represent the surface of a 3D object with a multi-scale mesh, re-triangulation is often a required processing step since arbitrary meshes do not usually exhibit multi-resolution connectivity. A 3D surface can be re-meshed through surface flattening (or parametrization). This type of technique maps the original surface to the complex plane [5]. The target domain of the map is easily equipped with a regular multi-resolution mesh. Thus, by interpolation, the spatial coordinates of the original surface are calculated at the vertices of that regular multi-scale grid. This methodology, used in [74] and Chapter II, allows us to equip each surface with a multi-resolution triangulation.

Now, in addition to their nice connectivity properties, multi-resolution mesh structures offer various opportunities for signal encoding. In particular, second generation wavelet transforms [87, 93] provide an efficient way to decompose a signal in both space and scale on a multi-resolution mesh, by projecting it onto a set of nested biorthogonal wavelet bases. The flexibility of these wavelets resides not only in their

capacity to deal with irregularly spaced data points but also in their adaptability to different kinds of supports (e.g. manifolds). Surface compression and smoothing constitutes the objective of the present work. Therefore, the signal that will be analyzed in this work consists of the spatial coordinates of the surface itself. Our methodology will build on the work of Nain *et al.* [74], which proposed encoding the spatial variations of a 3D surface using spherical wavelets. As any other wavelet representation, second generation wavelet schemes compress the essential part of a smooth signal into a fairly small set of large coefficients. Wavelet coefficients are stored by level of resolution and, for smooth signals, the distributions of coefficients tend to be very sparse in the finest levels (See Figure 3.14). Moreover, Figure 3.14 shows the impact of noise corruption on these distributions of coefficients at these same levels of resolution. Therefore, for any given surface, tremendous enhancements may be brought to the signal representation, not only by eliminating meaningless coefficients, but also by "de-noising" the observed coefficients. In this work, a wavelet-based de-noising model will be proposed that consists in removing the noisy part of each wavelet coefficient and keep the "clean" part of it. This procedure is referred to as *wavelet shrinkage*. Once this manipulation is performed, the inverse wavelet transform is applied to the modified set of coefficients and the de-noised signal can be observed in the original domain.

Many wavelet shrinkage models have been developed for first generation (or classical) wavelets. A quick overview of the major techniques is given in the following paragraphs, which will help the reader understand the context and the origins of the models developed in this thesis.

Wavelet shrinkage methodologies may be classified with respect to three different axes. These methods can be:

- **Thresholding or non-thresholding:** Non-thresholding shrinkage aims to

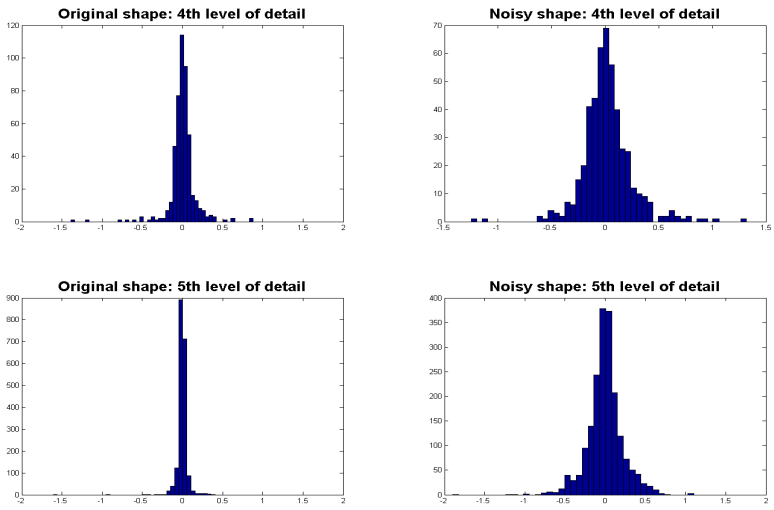


Figure 3.14: Distributions of coefficients from the fourth and fifth resolution levels for a typical spherical surface. We compare the distributions relative to the encoding of the smooth signal to that of a noisy shape. One may observe that the noiseless signal, which is aimed to be recovered by the de-noising model, is characterized by a large amount of low-valued coefficients

find an optimal noise-free value for each coefficient, whereas, with thresholding model, a coefficient is classified as either relevant or noisy, based on certain pre-defined criteria.

- **Adaptive or non-adaptive:** Non-adaptive shrinkage consists of a spatially uniform shrinkage rule that only takes into account the amplitude of the coefficient, whereas adaptive methods differentiate between coefficients based on the spatial context, the level of resolution, or singularity detection.

- **Classical or Bayesian:** Classical shrinkage manipulates coefficients individually without truly accounting for the overall distribution of a subset of coefficients. Bayesian methods, on the contrary, use the information one may have on the distribution of the wavelet coefficients. These methods are usually less ad-hoc and more robust than the classical models [98].

The most basic shrinkage techniques are **classical thresholding** rules. They apply a threshold to the whole set of wavelet coefficients and shrink those falling below that threshold. In [24], Donoho and Johnston propose defining a threshold for each resolution level as follows:

$$T_j = \sigma_j \cdot \sqrt{K \log(N_j)} \tag{3.23}$$

where $K = 2$, σ_j is the estimated noise standard deviation for the j-th resolution level and N_j is the number of coefficients in level j.

More sophisticated methods were next developed to make classical thresholding more **adaptive**, and many of these models were applied to the field of image de-noising. In [25], Donoho and Johnstone propose estimating a threshold that would minimize the Stein's unbiased risk. This was called the *Sureshrink* method. In [49, 75, 102], the threshold value is chosen such that the shrunk signal minimizes a reconstruction error estimated through cross-validation. Another type of threshold selection involves MSE (Mean Square Error) minimization after a prior model has been chosen for the wavelet coefficient distribution [17, 81]. In [77, 78], Ogden and Parzen propose using hypothesis testing in their threshold selection procedure. In this method, called *data-analytic* thresholding, coefficients are removed according to the significance of a statistical test, based on Brownian bridges sampling. Hypothesis testing also intervenes in the thresholding method developed in [2], where the false-discovery rate is introduced to account for the coefficients that have been kept whereas they should have been discarded.

All the aforementioned methods operate with a spatially uniform threshold rule, are very often too "universal" and seriously exhibit a lack of flexibility. Instead, if the threshold becomes spatially adaptive, performance in terms of smoothing and de-noising is significantly improved. In [43], the statistical hypothesis testing model proposed in [77,78] has been enhanced as it now takes into account both the magnitude of the coefficients and their spatial clustering properties. However, the threshold itself

remains spatially uniform. In [16, 18], however, Chang *et al.* propose adapting the value of the threshold based on the region of interest, using *context modeling*. Context modeling clusters wavelet coefficients according to their *context* specificities. Applications in image processing show how this significantly outperforms non-adaptive thresholding techniques. Another adaptive approach consists of considering the coefficients in overlapping blocks in order to incorporate information on neighboring coefficients into the wavelet shrinkage rule [13].

An alternative to these classical techniques is the development of **Bayesian** models. While results greatly depend on the quality of the threshold estimation in the classical shrinkage rules, Bayesian methods offer an opportunity to easily exploit and incorporate information on the coefficient distributions, to model the spread of the noise level, and to be less dependent on ad-hoc parameter estimations. Therefore, Bayesian models have often been proved to outperform classical shrinkage in the case of first-generation wavelets. Different types of Bayesian frameworks have been previously developed, and these are usually characterized by the way the shrinkage rule is implemented, by the method used to estimate the parameters of the model or by the nature of the distribution functions.

Pure Bayes rules may be used as shrinkers [90, 98] and usually exhibit nice properties for wavelet shrinkage. The shape of these shrinkers varies with the choice of prior distribution and the values of some model parameters. Alternative frameworks use Markov Random Fields (MRF) to encode both the prior and conditional probabilities of the Bayesian model. In MRF schemes, these two distributions are modeled as Gibbs probability functions [48, 69, 70, 79]. These adaptive techniques have become very popular in image de-noising. On the one hand, the MRF model allows the prior distribution to encode geometrical features of the image by spatially clustering coefficients (i.e. separating meaningful from meaningless coefficients). On the other hand, the conditional probabilities carry inter-scale information carried by coefficients from

coarser levels of resolution.

The intensity of shrinkage of these Bayesian frameworks depends on the quantity of noise that is assumed to be carried by the coefficient, but no notion of thresholding is ever introduced. These are instead referred to as *gradual* shrinkage methods. However, other Bayesian models have been developed that mimic the classical thresholding rule. In [3], thresholding is operated using posterior medians. The alternative proposed in [98] uses hypothesis testing as a thresholding tool. These thresholding-like frameworks allow the user to divide the set of coefficients into two categories. Larger and relevant coefficients may be kept intact or barely affected by the shrinkage, whereas noise-like coefficients may be shrunk to zero (hard thresholding) or reduced to a *noise-free* value (soft thresholding). In this context, hard thresholding frameworks fit best with the double objective of signal compression and de-noising since all the coefficients that are shrunk to zero can be "taken away" from the data set.

While several different models have been developed for classical wavelets, very little attention has been paid to shrinkage algorithms in the case of second generation wavelets. An extension to the MRF Bayesian shrinkage model has been proposed for image processing using second generation wavelets [20]. However, no thresholding-like algorithm has been specifically developed for shape analysis using spherical wavelets [87, 93]. This is the objective of the present work.

3.1.4 Contributions of This Work

This chapter presents the development of an empirical-based model operating as an **adaptive Bayesian thresholding**-like wavelet shrinkage rule for 3D shape signal de-noising and compression. The main contributions presented in this chapter may be listed as follows:

- **A novel empirical-based Bayesian thresholding-like wavelet shrinkage rule (Model I)**.

  First, the underlying structure of the shrinkage model is an extension to the concept presented in [98] for classical wavelets. The main contribution of this work resides in the fact that the proposed framework adapts to local specificities of the surface, while taking into account the characteristics of the spherical wavelet transform.

  Wavelet transforms tend to de-correlate signals, but usually, dependence subsists between a given coefficient and other close coefficients. Two types of correlation exist. First, a given coefficient may be correlated to its *parent coefficients*, i.e. to coefficients from coarser resolutions and for which the computation is influenced by this particular point. The set formed by a mesh node and all its parent vertices is referred to as its *cone of influence*. The variation rate in the amplitude of wavelets coefficients within a cone of influence usually translates the local regularity of the signal. The second type of correlation regards the neighborhood of the coefficient. In a smooth signal, the amplitude of a coefficient value is usually correlated to that of its neighbors. Relevant shape deformations are characterized by a certain local *activity* and can not be encoded by one isolated fine-scale coefficient. In the context of shape analysis, shape consistency imposes a relatively strong constraint on these two correlation factors.

  Therefore, in the context of 3D mesh analysis, one may assert that the information needed for assessing the relevance of a wavelet coefficient is carried by three different elements:

  - **Its amplitude**.
  - **The value of its parent coefficients (inter-scale information)**.

– **The spatial location of the carrying vertex on the mesh**.

The thresholding decision will be made according to these three factors. Details will be given in Section 3.4.2.

- **An efficient way to estimate the parameters in Model I, including noise level**.

  Several noise variance estimation methods have been developed for first generation wavelet shrinkage models [24, 98]. In Section 3.4.3.1, details are given on the methodology used to estimate this important parameter in the proposed model. We will also see that several other parameters require some particular attention.

- **An alternative Bayesian shrinkage framework is presented that models the spread of the noise level as a hyper-prior distribution function (Model II)**.

  In addition to the aforementioned model, this alternative Bayesian framework (presented in Sections 3.5 and 3.6) adds a supplementary level to the Bayesian rule as it models the spread of the noise variance and incorporates it as a hyper-prior distribution. This replaces the plugged-in noise value that needed to be estimated in the precedent model.

3.2 Proposed Shape Model and Wavelet Encoding

This section presents the model used to represent surfaces, as well as the wavelet transform scheme used to encode the shape signal.

3.2.1 Shape Model

In this work, surfaces are represented by 3D triangulated meshes. In most cases, these meshes have arbitrary structures that are very unlikely to offer multi-scale properties in their connectivity. Therefore, most shapes will be re-meshed in order to make feasible the multi-scale decomposition of the surface. In the present work, surface flattening is used to map the original mesh to a new regular multi-resolution grid [5, 74]. The multi-resolution structure is built by recursively subdividing a polyhedron [74], where each new subdivision consists of adding a finer *resolution level*. After j subdivisions, the refined grid contains $K(j)$ nodes. The $j+1^{th}$ subdivision introduces new nodes, which can be denoted by an index set $M(j)$. These new nodes are placed at the middle of each existing edge. Therefore, a subdivision splits each existing triangle into 4 new triangles. The complete set of nodes at the $j+1^{th}$ level of resolution is given by $K(j+1) = K(j) \bigcup M(j)$. The final mesh is made of N vertices, which are grouped by resolution level. The number of levels to be used will depend on the type of shape, and more levels will be needed for surfaces that exhibit higher curvature.

A given 3D surface can be represented by a $N \times 3$ matrix $[s]$ containing the (x, y, z) coordinates of the N vertices. As the noisy part of the signal is aimed to be removed, a surface will be modeled as follows:

$$[s] = [f] + [\varepsilon] \tag{3.24}$$

where $[f]$ is the $N \times 3$ matrix of the "noiseless" shape signal that we want to estimate, and $[\varepsilon]$ is assumed to be i.i.d. Gaussian noise. The entries of each vector are organized with respect to the *scaling* order of the vertices, starting with the vertices corresponding to the coarser resolution and ending with the highest resolution vertices.

This model is then encoded using wavelet decomposition.

3.2.2 Wavelet Encoding

Second generation wavelets [87, 93] enable any function to be decomposed and encoded on a multi-resolution mesh through projection onto a set of nested biorthogonal wavelet bases. A brief description of their structure and characteristics has been given in Chapter II.

The spherical wavelet transform can now be applied to the observed signal $[s]$ itself [74], which contains the 3D coordinates of all the surface vertices. Although the wavelet decomposition will be implemented using a Fast Discrete Wavelet Transform, the wavelet transform constitutes a linear filtering process and can be easily represented using matricial operations. Applying the wavelet filter to $[s]$ will consist in multiplying it by a matrix H, in which each column is a basis function evaluated at each one of the N vertices. Therefore, resulting from this simple multiplication, a matrix $[d]$ is obtained that translates the surface coordinates into a set of wavelet coefficients.

By the linearity of the wavelet transform, an additive representation of the shape is used in the wavelet domain:

$$[d] = [\theta] + [\eta] \tag{3.25}$$

where $[\theta]$ and $[\eta]$ are the $N\times 3$ matrices of wavelet coefficients that respectively encode the noiseless signal matrix $[f]$ and the noisy terms $[\varepsilon]$. The order in which the coefficients are ranked in these matrices is the same as in the matrices of the vertex coordinates, that is, from the coarsest level of subdivision to the finest. For notational purposes, $\underline{d}$ (respectively, $\underline{\theta}$) denotes a column vector from $[d]$ (respectively, $[\theta]$), which corresponds to the encoding of one of the three dimensions of the signal. Also, in the remainder of this chapter, $\mathbf{d}$ will refer to a vector of observed coefficients $[d_x, d_y, d_z]^T$ at an arbitrary vertex, and θ will similarly denote an arbitrary vector of signal part coefficients $[\theta_x, \theta_y, \theta_z]^T$.

3.3 Wavelet Thresholding Using Hypothesis Testing

As mentioned in the precedent paragraphs, our shape signal is composed of 3-dimensional vectors of coordinates -namely, one 3D vector per vertex. Thus, applying the wavelet transform to these vectors outputs vectors of wavelet coefficients $\mathbf{d} = [d_x, d_y, d_z]^T$. In the context of wavelet shrinkage, each coordinate could be assessed separately and independently, as proposed in [59]. However, in this model, correlation between the three dimensions of the signal is taken into account, and a multivariate shrinkage rule is applied to each vector of coefficients $\mathbf{d}$. The statistical model proposed in this work will therefore involve multivariate distributions of coefficients.

Given a triplet of coefficients $\mathbf{d}$, the proposed shrinkage model will aim to recover the corresponding noiseless coefficient θ. In order to mimic hard thresholding rules, the framework developed in this work is based on the evaluation of the following hypothesis:

$$H_0 : \theta = [\theta_x, \theta_y, \theta_z]^T = [0, 0, 0]^T \tag{3.26}$$

This hypothesis will be tested for each vector of wavelet coefficients, and the coefficients for which the hypothesis is "rejected" will be considered irrelevant to the signal representation. Concretely, the proposed hypothesis testing will be implemented within a Bayesian framework, and, for each triplet of coefficients $\mathbf{d}$, the following posterior odds will be estimated: $R = \frac{P(H_0|\mathbf{d})}{P(H_1|\mathbf{d})}$. If $R \geq 1$, then the vector of coefficients will be shrunk to zero, meaning that these coefficients are considered as noise. If $R < 1$, its observed value is kept and the given coefficient is accepted as part of the relevant part of the observed signal. This model therefore mimics hard-thresholding shrinkage rules. By shrinking noise-like coefficients to zero, it tends to simultaneously eliminate noise and compress the amount of information needed to encode the input shape signal.

The estimation of the aforementioned posterior odds requires the existence of a coherent Bayesian framework, for which prior and likelihood distributions need to

be carefully chosen and defined. Different approaches may be taken to model this framework. Not only several combinations of distribution functions may appropriately characterize the actual distribution of the wavelet coefficients, but also, the parameters of a given statistical model may carry important information and therefore require particular attention. The next two sections present the two models that this thesis proposes using. The main difference between these two models is the way the noise level is taken into account and incorporated into the framework. In the first case, an estimator is used and the noise variance values are plugged into the framework. In the second case, a hyper-prior distribution characterizes the spread of the noise variance itself. In both models, though, the coefficient prior and likelihood distributions are given similar structures.

3.4 Model I: Bayesian Wavelet Shrinkage with Plug-in Estimator for Noise Level

In this section, the first of the two models mentioned in Section 3.1.4 is described in detail. For the remainder of this chapter, this will be referred to as Model I. The main structure of the framework is first presented and the coefficient distribution functions are exhibited. Next, the model is modified in order to account for inter-scale and local signal information. Finally, the method used to estimate and incorporate the model parameters (including the noise level) into the framework is explained.

3.4.1 Structure of the Bayesian Framework

- **Prior**

The prior distribution is defined as a mixture of a point mass at zero (helpful for thresholding) and a spread distribution:

$$p(\theta) = \pi_0 \cdot \delta(0) + (1 - \pi_0) \cdot \xi(\theta), \tag{3.27}$$

where $\xi(\theta) \sim \mathcal{MVN}(0, \Phi)$, i.e.:

$$\xi(\theta) = \frac{1}{(2\pi)^{\frac{n}{2}} \mid \Phi \mid^{\frac{1}{2}}} \cdot \exp\left\{\frac{-\theta^T \cdot \Phi^{-1} \cdot \theta}{2}\right\} \tag{3.28}$$

where the covariance matrix Φ is estimated at each resolution level, and $n = 3$.

If using this prior distribution as is, the choice of a Bayesian model over the application of a simple thresholding technique would certainly not be worthwhile as the whole flexibility of such a framework is not used at its best. Moreover, the choice for a global value pi_0 would turn out to be crucial and the comments we have made about the risks of mis-calibrated smoothing for classical thresholding techniques would somehow apply here. This choice may indeed appear as arbitrary as that of a universal threshold value. This issue is addressed in Section 3.4.2.

- **Likelihood**

The likelihood distribution is represented by a multivariate normal distribution:

$$f(\mathbf{d}|\theta, \Sigma) = \frac{1}{(2\pi)^{n/2} \mid \Sigma \mid^{\frac{1}{2}}} \cdot \exp\left(-\frac{(\mathbf{d}-\theta)^T \Sigma^{-1} (\mathbf{d}-\theta)}{2}\right) \tag{3.29}$$

where Σ, the noise covariance matrix, is estimated by $\widehat{\Sigma}$ (see Section 3.4.3). The assumption is made that Σ is diagonal as we assume the three components (x, y and z) of the noise to be uncorrelated:

$$\Sigma = \begin{vmatrix} \sigma_x^2 & 0 & 0 \\ 0 & \sigma_y^2 & 0 \\ 0 & 0 & \sigma_z^2 \end{vmatrix}.$$

For notational purposes, we will write $f(\mathbf{d}|\theta)$ instead of $f(\mathbf{d}|\theta, \Sigma)$ in the remainder of this section.

- **Marginal**

The marginal distribution may be written as follows:

$$\begin{aligned} p(\mathbf{d}) &= \int f(\mathbf{d}|\theta) \cdot p(\theta) d\theta \\ &= \pi_0 \cdot f(\mathbf{d}|\mathbf{0}) + (1 - \pi_0) \cdot \int \xi(\theta) f(\mathbf{d}|\theta) d\theta \end{aligned} \tag{3.30}$$

In Equation (3.30), let us define $I = \int \xi(\theta) f(\mathbf{d}|\theta) d\theta$ (second term on the right side of the equation). I may be expanded as follows:

$$I = \int \frac{1}{(2\pi)^3 \mid \Phi \mid^{\frac{1}{2}} \mid \Sigma \mid^{\frac{1}{2}}} \cdot \exp\left\{ -\frac{\theta^T \Phi^{-1} \theta + (\mathbf{d} - \theta)^T \Sigma^{-1} (\mathbf{d} - \theta)}{2} \right\} d\theta \tag{3.31}$$

Let Σ_*^{-1} be such that $\Sigma^{-1} = \Sigma_*^{-1} \cdot \Sigma_*^{-1}$. Note that $(\Sigma_*^{-1})^T = \Sigma_*^{-1}$. Thus, the precedent equation can be written:

$$I = \int \frac{1}{(2\pi)^3 \mid \Phi \mid^{\frac{1}{2}} \mid \Sigma \mid^{\frac{1}{2}}} \cdot \exp\left\{ -\frac{\theta^T \Phi^{-1} \theta + (\mathbf{d} - \theta)^T (\Sigma_*^{-1})^T \Sigma_*^{-1} (\mathbf{d} - \theta)}{2} \right\} d\theta$$

One may operate the change of variable $\theta \leftarrow \Sigma_*^{-1}\theta$. Also, let us define $\mathbf{d}^* = \Sigma_*^{-1}\mathbf{d}$ and $(\Phi^*)^{-1} = \Sigma_*^T \Phi^{-1} \Sigma_*$. Therefore, the following holds:

$$\begin{aligned} I &= \int \frac{1}{(2\pi)^3 \mid \Phi \mid^{\frac{1}{2}} \mid \Sigma \mid^{\frac{1}{2}}} \cdot \exp\left\{ -\frac{\theta^T (\Phi^*)^{-1} \theta + (\mathbf{d}^* - \theta)^T (\mathbf{d}^* - \theta)}{2} \right\} \mid \Sigma \mid^{\frac{1}{2}} d\theta \\ &= \frac{1}{(2\pi)^3 \mid \Phi \mid^{\frac{1}{2}}} \cdot \int \exp\left\{ -\frac{Q}{2} \right\} d\theta \end{aligned} \tag{3.32}$$

where $Q = \theta^T (\Phi^*)^{-1} \theta + (\mathbf{d}^* - \theta)^T (\mathbf{d}^* - \theta)$.

Using usual results on multivariate integration, I can be explicitly expanded:

$$I = \frac{\mid ((\Phi^*)^{-1} + I_d)^{-1} \mid^{\frac{1}{2}}}{(2\pi)^{\frac{3}{2}} \mid \Phi \mid^{\frac{1}{2}}} \cdot \exp\left\{ -\frac{(\mathbf{d}^*)^T [I_d - ((\Phi^*)^{-1} + I_d)^{-1}] \mathbf{d}^*}{2} \right\} \tag{3.33}$$

- **Posterior and hypothesis testing**

For each triplet of coefficients, one may now evaluate the posterior probability $P(H_0|\mathbf{d})$ as follows:

$$P(H_0|\mathbf{d}) = P(\theta = 0|\mathbf{d}) = \frac{P(\mathbf{d}|\theta = 0) P(\theta = 0)}{p(\mathbf{d})}$$

Therefore, using (3.30), the posterior probability may be written as follows:

$$P(H_0|\mathbf{d}) = \frac{\pi_0 \cdot \frac{1}{(2\pi)^{n/2}|\Sigma|^{\frac{1}{2}}} \cdot \exp\left(\frac{-\mathbf{d}^T\Sigma^{-1}\mathbf{d}}{2}\right)}{\pi_0 \cdot \frac{1}{(2\pi)^{n/2}|\Sigma|^{\frac{1}{2}}} \cdot \exp\left(\frac{-\mathbf{d}^T\Sigma^{-1}\mathbf{d}}{2}\right) + (1-\pi_0) \cdot \frac{|((\Phi^*)^{-1}+I_d)^{-1}|^{\frac{1}{2}}}{(2\pi)^{\frac{3}{2}}|\Phi|^{\frac{1}{2}}} \cdot \exp\left(-\frac{(\mathbf{d}^*)^T[I_d-((\Phi^*)^{-1}+I_d)^{-1}]\mathbf{d}^*}{2}\right)} \tag{3.34}$$

3.4.2 An Adaptive Bayesian Framework

The model that has been described in the above section would assess each coefficient separately, and make a thresholding decision by only considering its amplitude without taking into account the neighborhood of the corresponding vertex. As explained in Section 3.1.4, correlation exists between wavelet coefficients and relevant shape deformations are characterized by a certain local *activity*. Therefore, these can not be encoded by one isolated fine-scale coefficient. Instead, relevant local shape information will need a much larger support, while isolated fine-scale coefficients will be more likely regarded as noisy artifacts. Two major elements will enhance the original model.

On the one hand, for any given wavelet coefficient from level j, it is desirable to account for the information brought by its parent coefficients. Indeed, even though the wavelet transform *de-correlates* the signal by decomposing it across several scale levels, dependence exists between a given coefficient and its parent coefficients from coarser levels. The set formed by any mesh point and its neighboring points from coarser resolution for which the wavelet transform is influenced by this particular point is referred to as its *cone of influence*. The evolution of the amplitude of wavelets coefficients through the cone of influence at a particular spatial location usually translates the local regularity of the signal around this area. *Persistence* exists across scales for signal part coefficients where noise terms will not find such support from coarser levels. This particularity is used to update the thresholding rule: The higher the average

value of its parent coefficients, the more chance a coefficient has to be kept.

On the other hand, one could argue that the location of mesh vertices on a surface is quite independent of the surface variations, as the re-meshing step may have assigned vertices regardless of the shape topology. Thus, referring to neighbor vertices only may not provide complete information on the surface area surrounding the vertex at which we are assessing the wavelet coefficients. In fact, the coefficients defined at its neighboring vertices may only characterize a very localized and irrelevant alteration of the original shape, whereas the surrounding area may be quasi flat overall. Thus, by incorporating a curvature term κ into our model, the shrinkage rule is strengthened when local curvature is low. This curvature term, when close to zero, will tend to attenuate the influence of high valued neighboring coefficients. In order to meet this objective, a decision needs to be made on the nature and computation of κ. κ may be defined as follows: $\kappa = \max(|\kappa_1|, |\kappa_2|)$, where κ_1 and κ_2 are estimates of the two principal curvatures at a given vertex. In order to estimate these two principal curvatures, one may use the technique proposed in [95], which describes a simple way to estimate shape curvature on a triangulated mesh. Since the observed signal may exhibit some irregular noisy artifacts, the principal curvatures may be inaccurately estimated. Therefore, the coefficients from several levels of resolution are temporarily set to zero. This action is referred to as *linear shrinkage*. By doing this, one obtains a very smooth approximation of our surface and a much easier way to compute curvature estimation over the mesh. Of course, this linear shrinkage over-smoothes the shape as it removes some of the very local shape variations, but this method still allows one to reasonably characterize the major features of the curvature term over the surface. In Section 3.6, a discussion is given on this curvature estimation and explains how this may provide the end-user with the capability to control the intensity of the smoothing.

As both parent coefficients and curvature are considered in the shrinkage rule,

both sources of information need to be balanced by strategically mixing them in the body of our prior distribution. Given the structure of the prior distribution, we have incorporated all this additional information into the mixture weight, now denoted by $\widetilde{\pi}_0$. Instead of being instantiated with a simple global value, $\widetilde{\pi}_0$ now varies with the spatial location of the assessed coefficient. It becomes a function of the parent coefficients' value and of local curvature. Overall, $\widetilde{\pi}_0$ should be a function taking values in $[0, 1]$ and should decrease with respect to local curvature and the amplitude of parent coefficients -$\widetilde{\pi}_0$ needs to be close to 1 when the neighborhood does not exhibit any relevant pattern. For a vector θ that is to be estimated at a vertex of level j, one may define $\widetilde{\pi}_0$ as follows:

$$\widetilde{\pi}_0(\theta) = K \cdot \exp\{-\beta \cdot \bar{\kappa} \cdot F(\theta)\} \tag{3.35}$$

where $\bar{\kappa}$ ($\in [0, 1]$) stems from normalizing κ, K is a constant, and $F(\vec{\theta})$ is a function of the parent coefficients. Given a coefficient θ_x (respectively θ_y, θ_z) encoding the x (respectively y, z) coordinate of the signal at an arbitrary vertex, $C(\theta_x)$ (respectively $C(\theta_y)$, $C(\theta_z)$) denotes the average value of its direct parent coefficients. Then, one may impose that parent coefficients should not influence thresholding if their average $C(\theta_x)$ (respectively $C(\theta_y)$, $C(\theta_z)$) falls under a certain threshold T_x (respectively T_y, T_z). Therefore, F may be defined as follows:

$$F(\theta) = max\{(C(\theta_x) - T_x)_+, (C(\theta_y) - T_y)_+, (C(\theta_z) - T_z)_+\} \tag{3.36}$$

where $(U)_+ = max(U, 0)$. Thus, F will be close to zero when the average of parent coefficients is low-valued, i.e. when no real pattern is observed locally (i.e. across the relative cone of influence).

With this definition of $\widetilde{\pi}_0$, the new adaptive prior distribution becomes:

$$p(\theta) = \widetilde{\pi}_0 \cdot \delta(0) + (1 - \widetilde{\pi}_0) \cdot \xi(\theta) \tag{3.37}$$

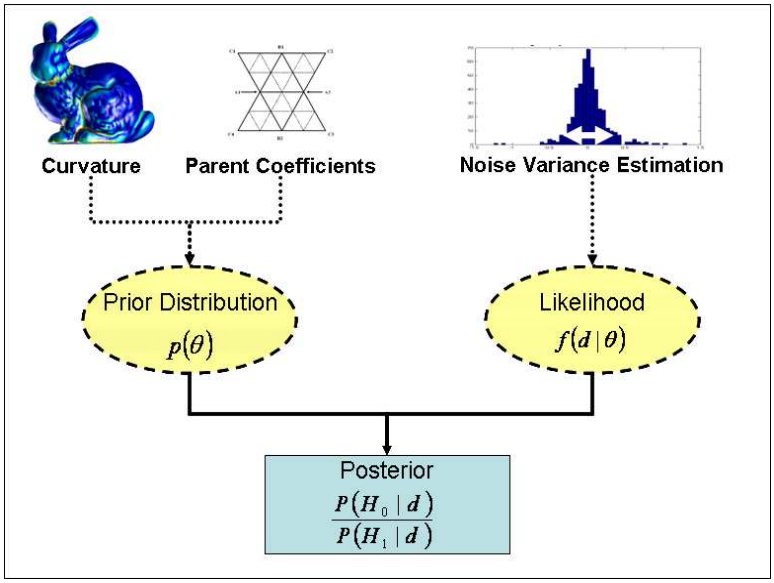


Figure 3.15: Structure and sub-elements of the proposed Bayesian framework.

Therefore, the posterior probability may be written as follows:

$$P(H_0|\mathbf{d}) = \frac{\widetilde{\pi}_0 \cdot \frac{1}{(2\pi)^{n/2}|\Sigma|^{\frac{1}{2}}} \cdot \exp\left(\frac{-\mathbf{d}^T\Sigma^{-1}\mathbf{d}}{2}\right)}{\widetilde{\pi}_0 \cdot \frac{1}{(2\pi)^{n/2}|\Sigma|^{\frac{1}{2}}} \cdot \exp\left(\frac{-\mathbf{d}^T\Sigma^{-1}\mathbf{d}}{2}\right) + (1-\widetilde{\pi}_0) \cdot \frac{|((\Phi^*)^{-1}+I_d)^{-1}|^{\frac{1}{2}}}{(2\pi)^{\frac{3}{2}}|\Phi|^{\frac{1}{2}}} \cdot \exp\left(-\frac{(\mathbf{d}^*)^T[I_d-((\Phi^*)^{-1}+I_d)^{-1}]\mathbf{d}^*}{2}\right)} \tag{3.38}$$

The complete Bayesian framework is graphically summarized in Figure 3.15.

3.4.3 Parameter Estimation

In this proposed model, both Σ and Φ need to be estimated for each level of resolution.

3.4.3.1 Estimator for Σ

$\widehat{\Sigma}$ is the estimator for the noise covariance matrix. Noise is assumed to be isotropic (i.e. the variance is the same for all three dimensions) and the three dimensions of the noise are assumed to be independent (i.e. $\widehat{\Sigma}$ is diagonal). Several techniques could be used to estimate $\widehat{\Sigma}$. In [59], a power spectrum-based technique is proposed that takes into account the decay of power across the successive levels of resolution. This works

fine for a certain category of shapes but estimating the slope of the noiseless curve in the *scalogram* may become challenging in some other cases. Another method has been proposed to estimate Σ that one can decomposed into two steps.

A first rough estimate of Σ for each level of resolution is done using the "known" noise power ratios between successive resolution levels. Indeed, by projecting the shape signal on wavelet bases that have been built on the shape itself, experiments show that, in the wavelet domain, the ratios between the noise variance of two successive resolution levels appear to be intrinsic to the structure of the wavelet scheme and independent of the overall noise corruption of the shape. Thus, after the noise variance of the finest level of resolution has been estimated, we are able to successively estimate the noise variance of all coarser levels. Given the noise variance of level $j+1$, we simply use the corresponding ratio and estimate the variance for level j: This step is repeated down the scale of resolution. In order to estimate the noise variance at the finest level of resolution, the multi-resolution mesh is artificially upsampled through the addition of one extra level, in which all wavelet coefficients are considered irrelevant high frequency artifacts (i.e. noise). Thus, estimating the noise variance for this finest level of resolution consists in computing the variance of the observed coefficients of this level.

Once these estimates have been found, adjustments are made through coefficient distribution matching. As derived in section 3.4.1, $p(\mathbf{d})$ is a function of $\widehat{\Sigma}$. If the model is appropriate and the parameters well-chosen, this distribution should match the empirical distribution of observed coefficients for each level of resolution. Therefore, the objective consists in finding the optimal $\widehat{\Sigma}$, which would provide the best fit between $p(\mathbf{d})$ and the empirical coefficient distribution. To do so, we tune the noise variance within a reasonable interval of amplitudes defined around its initial estimate, and, for each of these different $\widehat{\Sigma}$, the goodness of fit between $p(\mathbf{d})$ and the empirical

distribution of the observed coefficients is evaluated. In order to do so, we can use *chi-square goodness-of-fit* testing, where the null hypothesis is "$p(\mathbf{d})$ and the distribution of the observed coefficients are similar". The value of $\widehat{\Sigma}$ for which the test statistic is the smallest corresponds to our final estimate of Σ for the given level of resolution.

3.4.3.2 Estimator for Φ

At each level of resolution, Φ, the covariance matrix of the noiseless signal θ is estimated based on $\widehat{\Sigma}$. Indeed, given the linearity of the problem described in Equation (3.25) and the independence of $[\eta]$ and $[\theta]$, one may write:

$$\widehat{\Phi} = (\Xi - \widehat{\Sigma})/(1 - \bar{\pi_0})^2 \tag{3.39}$$

where $\widehat{\Phi}$ is the estimator for Φ, Ξ is the covariance matrix of the observed coefficients $[d]$ at the given level, and $\bar{\pi_0}$ is the average value of $\widetilde{\pi}_0$ for the level of resolution under investigation. The division by $(1 - \bar{\pi_0})^2$ is due to the weighted mixture of the prior $p(\theta)$.

These estimators provide decent values for Φ and Σ and Section 3.6 shows that the entire model remains very robust to these estimations. However, the Bayesian paradigm offers an opportunity to go further in the modeling of the noise variance. Indeed, the spread of the noise level may be represented as an hyper-prior distribution function and integrated into the main Bayesian framework. This is the main add-on of the alternative method that is presented in the next section.

3.5 Model II: Bayesian Wavelet Shrinkage with Normal Inverse Gamma Hyper-prior

In this section, an alternative wavelet shrinkage model is described, where an hyper-prior distribution is introduced in order to characterize the spread of the noise variance. Similarly to the precedent section, the coefficient distribution functions are

presented and the different assumptions that have been made are explained. Further details are given on the effect of the hyper-prior on the overall model.

3.5.1 Structure of the Bayesian Framework

- **Prior**

Again, the prior distribution is defined as a mixture of a point mass at zero and a spread distribution:

$$p(\theta|\sigma^2) \sim \pi_0 \cdot \delta_0 + (1-\pi_0) \cdot \mathcal{MVN}(0, \sigma^2\Phi) \tag{3.40}$$

where σ^2 is the noise variance and Φ will be estimated as it represents the 3×3 covariance matrix for θ.

A prior is defined for σ^2. This is chosen to be an Inverse Gamma distribution:

$$p(\sigma^2) = \frac{(\alpha/2)^{\delta/2} \cdot \exp\left(-\frac{\alpha}{2\cdot\sigma^2}\right)}{\Gamma(\delta/2) \cdot \sigma^{2(\frac{\delta+2}{2})}} \tag{3.41}$$

where α and δ are hyperparameters to be estimated.

Therefore, we obtain:

$$p(\theta, \sigma^2) = \frac{(\alpha/2)^{\delta/2} \cdot \exp\left(-\frac{\alpha}{2\cdot\sigma^2}\right)}{\Gamma(\delta/2) \cdot \sigma^{2(\frac{\delta+2}{2})}} \cdot \left(\pi_0 \cdot \delta_0 + \frac{(1-\pi_0)}{(2\pi\sigma^2)^{n/2}|\Phi|^{1/2}} \cdot \exp\left(-\frac{\theta^T\Phi^{-1}\theta}{2\sigma^2}\right)\right) \tag{3.42}$$

By integrating out σ^2, the "marginal" distribution for θ may be written:

$$\begin{aligned} p(\theta) &= \int p(\theta, \sigma^2) d\sigma^2 \\ &= \pi_0 \cdot \delta_0 + (1-\pi_0) \cdot \frac{(\alpha/2)^{\delta/2}}{\Gamma(\delta/2)(2\pi)^{n/2}|\Phi|^{1/2}} \int (\sigma^2)^{(-\frac{\delta+n+2}{2})} \cdot \exp\left(-\frac{\theta^T\Phi^{-1}\theta + \alpha}{2\sigma^2}\right) d\sigma^2 \\ &= \pi_0 \cdot \delta_0 + \pi_1 \cdot \frac{\alpha^{\delta/2}\Gamma(\frac{\delta+n}{2})}{\Phi|^{1/2}\pi^{n/2}\Gamma(\delta/2)} \cdot \left(\theta^T\Phi^{-1}\theta + \alpha\right)^{(-\frac{\delta+n}{2})} \\ &\sim \pi_0 \cdot \delta_0 + (1-\pi_0) \cdot t_\delta(\mathbf{0}, \alpha\Phi) \end{aligned} \tag{3.43}$$

where $t_\delta(\mathbf{0}, \alpha\Phi)$ is the multivariate t-distribution with δ degrees of freedom.

- **Likelihood**

The likelihood distribution is represented by a multivariate normal distribution:

$$f(\mathbf{d}|\theta,\sigma^2) = \frac{1}{(2\pi\sigma^2)^{n/2}} \cdot \exp\left(\frac{-(\mathbf{d}-\theta)^T(\mathbf{d}-\theta)}{2\sigma^2}\right) \tag{3.44}$$

- **Marginal**

The marginal distribution may be written as follows:

$$p(\mathbf{d}) = \int f(\mathbf{d}|\theta,\sigma^2) \cdot p(\theta|\sigma^2) d\sigma^2 d\theta \tag{3.45}$$

After derivation ,we obtain:

$$p(\mathbf{d}) = \frac{\alpha^{(\delta/2)}\Gamma(\delta^*/2)}{\pi^{n/2}\Gamma(\delta/2)} \cdot \left(\pi_0 \cdot (\mathbf{d}^T\mathbf{d}+\alpha)^{(-\delta^*/2)} + (1-\pi_0) \cdot \frac{|\Phi^*|^{1/2}}{|\Phi|^{1/2}} (\alpha^*)^{(-\delta^*/2)}\right) \tag{3.46}$$

where

$$\Phi^* = (\Phi^{-1} + I)^{-1} \tag{3.47}$$

$$\delta^* = \delta + n \tag{3.48}$$

$$\alpha^* = \alpha + \mathbf{d}^T(I - \Phi^*)\mathbf{d} \tag{3.49}$$

- **Posterior and hypothesis testing**

For each triplet of coefficients, we can now evaluate the posterior probability $P(H_0|\mathbf{d})$ as follows:

$$P(H_0|\mathbf{d}) = P(\theta = 0|\mathbf{d}) = \int P(\theta = 0, \sigma^2|\mathbf{d}) d\sigma^2 \tag{3.50}$$

where

$$\begin{aligned} P(\theta = 0, \sigma^2|\mathbf{d}) &= \frac{P(\mathbf{d}|\theta = 0, \sigma^2) P(\theta = 0, \sigma^2)}{p(\mathbf{d})} \\ &= \frac{1}{p(\mathbf{d})} \cdot \frac{\pi_0 \cdot (\alpha/2)^{\delta/2} \exp\left(\frac{-\alpha}{2\sigma^2}\right)}{\Gamma(\delta/2)\sigma^{2(\frac{\delta+2}{2})}} \cdot \frac{1}{(2\pi\sigma^2)^{n/2}} \cdot \exp\left(\frac{-\mathbf{d}^T\mathbf{d}}{2\sigma^2}\right) \end{aligned}$$

Using (3.46), we may therefore write:

$$P(\theta = 0, \sigma^2|\mathbf{d}) = \frac{\pi_0 \cdot \sigma^{-2(\frac{n+\delta+2}{2})} \exp\left(\frac{-\mathbf{d}^T\mathbf{d}+\alpha}{2\sigma^2}\right)}{2^{(\delta^*/2)}\Gamma(\delta^*/2) \cdot \left(\pi_0 \cdot (\mathbf{d}^T\mathbf{d}+\alpha)^{(-\delta^*/2)} + (1-\pi_0) \cdot \frac{|\Phi^*|^{1/2}}{|\Phi|^{1/2}} (\alpha^*)^{(-\delta^*/2)}\right)}$$

Thus,

$$P(H_0|\mathbf{d}) = \frac{\pi_0 \cdot \int \exp\left(\frac{-\mathbf{d}^T\mathbf{d}+\alpha}{2\sigma^2}\right)\sigma^{-2(\frac{n+\delta+2}{2})}d\sigma^2}{2^{(\delta^*/2)}\Gamma(\delta^*/2)\cdot\left(\pi_0\cdot(\mathbf{d}^T\mathbf{d}+\alpha)^{(-\delta^*/2)}+(1-\pi_0)\cdot\frac{|\Phi^*|^{1/2}}{|\Phi|^{1/2}}(\alpha^*)^{(-\delta^*/2)}\right)}$$
$$= \frac{\pi_0\cdot(\mathbf{d}^T\mathbf{d}+\alpha)^{(-\delta^*/2)}}{\left(\pi_0\cdot(\mathbf{d}^T\mathbf{d}+\alpha)^{(-\delta^*/2)}+(1-\pi_0)\cdot\frac{|\Phi^*|^{1/2}}{|\Phi|^{1/2}}(\alpha^*)^{(-\delta^*/2)}\right)} \quad (3.51)$$

3.5.2 An Adaptive Bayesian Framework

Similarly to the precedent model, this framework may account for inter-scale correlation as well as local consistency, as described in Section 3.4.2. This dual information may be carried by the prior mixture weights π_0. The statistical model therefore become adaptive. For any coefficient vector θ, one may define $\widetilde{\pi}_0$ as shown in (3.35).

With this definition of $\widetilde{\pi}_0$, the new adaptive prior distribution becomes:

$$p(\theta|\sigma^2) \sim \widetilde{\pi}_0\cdot\delta_0 + (1-\widetilde{\pi}_0)\cdot\mathcal{MVN}(0,\sigma^2\Phi) \quad (3.52)$$

Therefore, the posterior probability may be written as follows:

$$P(H_0|\mathbf{d}) = \frac{\widetilde{\pi}_0\cdot(\mathbf{d}^T\mathbf{d}+\alpha)^{(-\delta^*/2)}}{\left(\widetilde{\pi}_0\cdot(\mathbf{d}^T\mathbf{d}+\alpha)^{(-\delta^*/2)}+(1-\widetilde{\pi}_0)\cdot\frac{|\Phi^*|^{1/2}}{|\Phi|^{1/2}}(\alpha^*)^{(-\delta^*/2)}\right)} \quad (3.53)$$

3.5.3 Parameters

3.5.3.1 Estimators for α and δ

α and δ are related to each other via $\mathbb{E}(\sigma^2) = \frac{\alpha}{\delta-2}$. Let $\widehat{\sigma}^2$ be an estimator for $\mathbb{E}(\sigma^2)$. Thus, the following approximation is used: $\widehat{\sigma}^2 = \frac{\alpha}{\delta-2}$. In order to find good estimators for α,δ and $\mathbb{E}(\sigma^2)$, coefficient distribution matching is implemented. $p(\mathbf{d})$ is a function of α, δ and $\widehat{\sigma}^2$. If the model is appropriate and the parameters well-chosen, this distribution should match the empirical distribution of observed coefficients for each level of resolution. Therefore, the objective consists in finding the optimal α,δ and $\widehat{\sigma}^2$, which would provide the best fit between $p(\mathbf{d})$ and the empirical coefficient distribution. As described in Section 3.4.3.1, adequate values are found for these

parameters by optimizing the "goodness-of-fit" between the two distributions. In order to evaluate the goodness of that fit, we used 3-dimensional *chi-square goodness-of-fit* testing, where the null hypothesis is "$p(\mathbf{d})$ and the distribution of the observed coefficients are similar". Thus, we first divide the data space into 3d bins. The number of observed coefficients falling in each bin i is denoted by $O(i)$, and the theoretical value $p(\mathbf{d} \in \text{Bin i})$ is referred to as $E(i)$ (expected frequency for bin i). Finally, we compute the following statistic:

$$\chi^2 = \Sigma \frac{(O(i)^2 - E(i)^2)}{E(i)} \tag{3.54}$$

Then, we choose the vector [α,δ] that corresponds to the lowest statistic χ^2.

3.5.3.2 Estimators for Φ

According to the distribution $p(\theta|\sigma^2)$ (Equation (3.40)), we can write:

$$var(\theta) = \mathbb{E}(var(\theta|\sigma^2)) + var(\mathbb{E}(\theta|\sigma^2)) = (1-\pi_0)^2 \cdot \mathbb{E}(\sigma^2) \cdot \Phi \tag{3.55}$$

As explained in Section 3.4.3.2, this covariance matrix can also be expanded as follows:

$$var(\theta) = (\Xi - \Sigma) \tag{3.56}$$

where Ξ is the covariance matrix of the observed coefficients [d] and Σ is an isotropic diagonal matrix with $\mathbb{E}(\sigma^2)$ at all three entries of its diagonal.

Using Equations (3.55) and (3.56), we obtain:

$$\Phi = \frac{\delta - 2}{(1-\pi_0)^2 \cdot \alpha} \cdot (\Xi - \Sigma) \tag{3.57}$$

Thus, using the computed estimates of $\mathbb{E}(\sigma^2)$, α, and δ, Φ can be reasonably estimated.

Table 3.1: Compression Rates and Reconstruction Errors

		BUNNY	CAUDATE	HELICOPTER	HIPPOCAMPUS
# of Coefficients		10242	2562	1026	2562
Compression Rates	Model I	68%	75%	90%	70%
	Model II	52%	75%	80%	68%
Mean Error (% of the bounding box)	Model I	0.18%	0.55%	2.5%	0.31%
	Model II	0.16%	0.49%	2.2%	0.3%
Max Error (% of the bounding box)	Model I	4%	3.6%	4%	4.2%
	Model II	3.9%	3.6%	4%	4.2%

3.6 Experiments and Analysis

3.6.1 Protocol

The two proposed models (Sections 3.4 and 3.5) have been tested on several different shapes: BUNNY [1], HELICOPTER (2 sample shapes), HIPPOCAMPUS (set of 25 shapes), CAUDATE (set of 27 shapes). All have a spherical topology and have been re-meshed as described in Section 3.1.4. The number of resolution levels used for multi-scale decomposition varies with the complexity of the shape and the precision provided by the phase of data acquisition. More precisely, for a given original mesh, we select the multi-scale level of resolution of the regular mesh as the first one that has more vertices than the original one. This prevents from loosing information during the re-triangulation phase without over-sampling the data either. The BUNNY mesh contains seven resolution levels, the HIPPOCAMPUS and HELICOPTER, six and the CAUDATE, five. Furthermore, to test the robustness to noise of the method, Gaussian noise has been added to original shapes and the proposed shrinkage has also been applied to the artificially corrupted shapes.

For both models, the weights $\widetilde{\pi}_0$ of the prior distribution are computed for each coefficient after F_θ has been evaluated. The function F_θ has three parameters that

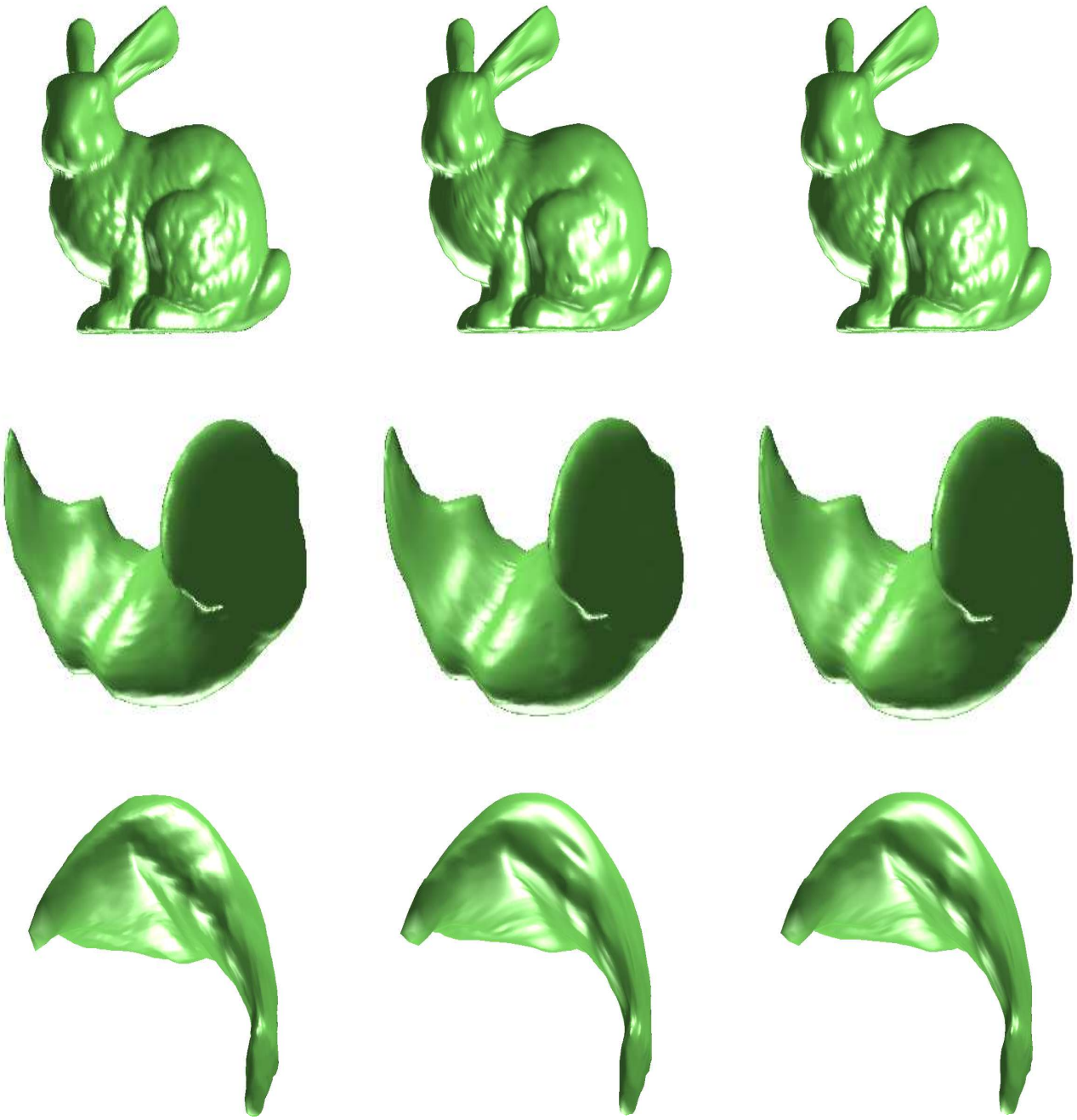
require to be instantiated: T_x, T_y and T_z. One can simply choose them equal to zero. This way, there is no "plateau" in the F function as any increment in the amplitude of the parent coefficients tends to attenuate the shrinkage. The profile of $\widetilde{\pi}_0$ also depends on parameter β. This parameter can be set by deciding on the limit of $\widetilde{\pi}_0$ as F_θ gets large. It has been decided that $\widetilde{\pi}_0$ would fall under a value close to 0 as soon as the parent coefficients get values that are larger than the average value of the corresponding standard deviation. Thus, if the average value of parent coefficients gets higher than the corresponding standard deviation, the high chance of keeping a coefficient will certainly be due to high valued parents. Finally, another parameter that need be estimated is K. This constant represents the maximal value of $\widetilde{\pi}_0$ and gets updated at each resolution level. At a given level, K is equal to the percentage of coefficients that fall below the universal threshold [24].

3.6.2 A Double Objective: Compression and Smoothing

Through these experiments, we have observed how the two proposed methods combine compression and smoothing and how it enables one to recover shape signal with more consistency than other methods.

3.6.2.1 A Compression Technique

Compression is easily quantifiable by computing the percentage of coefficients that are shrunk to zero when applying thresholding. The compression rates obtained with the proposed methods on the aforementioned shapes are presented and compared in Table 3.1. A great majority of the wavelet coefficients are taken away. In the first row, the number of coefficients that the proposed algorithm has been applied to and from which the compression rates are computed. It may not contain the total number of vertices that are used to represent the shape. Indeed, for the BUNNY shape, the multi-resolution mesh contains a total of 40,962 vertices. The finest level of definition is so insignificant in the shape representation that we have decided to



(a) (b) (c)

Figure 3.16: Compression of shape signals. First row: Results for the BUNNY, Second row: Results for the HIPPOCAMPUS, Third row: Results for the CAUDATE. (a): Original shape, (b): Smoothed Shape with Model I, (c): Smoothed Shape with Model II.

linearly shrink it. Linear shrinkage consists of strictly zeroing all coefficients of the level. This can happen when the resolution of the multi-scale mesh obtained after the re-meshing phase inappropriately high. In this case, the shape gets over-sampled and the finest levels of the mesh end up being helpless for signal encoding. If one had included these finest levels in the computation of the compression rate, these rates would have obviously been even larger. In the *Compression Rates* row, the percentage of coefficients that have been shrunk, out of the aforementioned total number of coefficients, is shown. These results are presented for both methods. On average, more than 70% of the processed coefficients are shrunk to zero without any significant loss of relevant information. The *Error* row shows the reconstruction error as a percentage of the bounding box (smaller axis). This loss can be estimated by computing the L2 distance between the original shape and the resulting de-noised signal. For the HELICOPTER shape, the error may seem to be higher than for other shapes, but this can be explained by the difference of length between the longer and the smaller axis. Besides this, the error values are very reasonable. Model II provides better error rates than Model I.

In Figure 3.16, results of wavelet thresholding are presented for three different shapes: BUNNY, HIPPOCAMPUS and CAUDATE. This shows how the signal is preserved despite the high compression rates.

3.6.2.2 A Smoothing and De-noising Technique

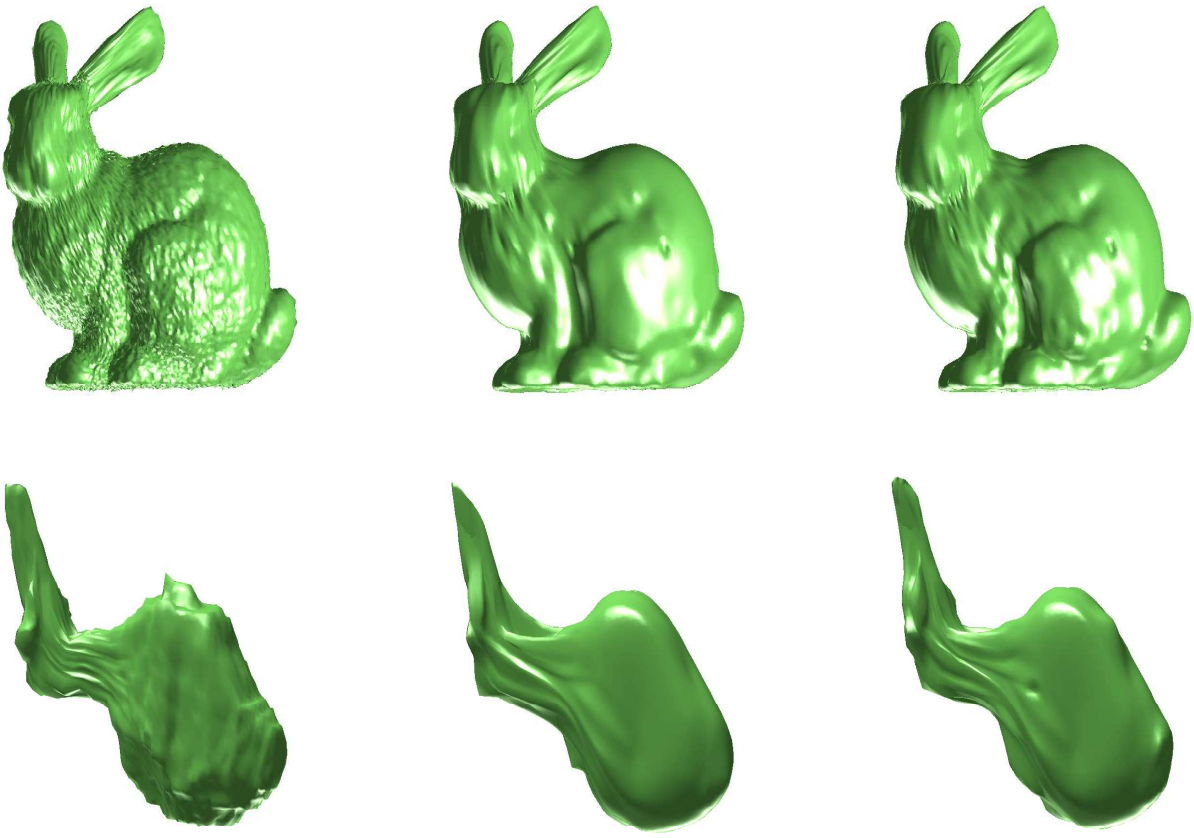
Smoothing, as opposed to compression, remains a subjective concept that is often difficult to evaluate. In the proposed model, smoothing can always be controlled by adjusting the value of the curvature term κ and/or the structure of the inter-scale information. First, κ is computed on an artificially smoothed version of the surface after several levels of resolution have been temporarily removed (linear shrinkage). If the number of levels that are linearly shrunk changes, the distribution of κ may

slightly vary. Also, at any given vertex, the model does not incorporate the "raw" value of κ. Instead, in order to carry better information about the local curvature, κ is averaged out over a pre-defined set of neighboring vertices. The size of this neighboring region may slightly affect the final value of κ. Finally, as inter-scale information is incorporated into the framework, the number of parent coefficients that are taken into account will have an effect on the shrinkage rule. For all these reasons, different adjustments have been made and tested for smoothing. Only some of these results are presented in this thesis.

The proposed algorithm has been applied to various shapes in order to test the smoothing impact of the model. Also, to see how the proposed algorithm performs on more noisy signals, some of those shapes have been artificially corrupted with noise and then "de-noised" using the proposed method. In order to estimate κ at each vertex of a given level of resolution J, the levels finer than J are linearly shrunk.

Figure 3.17 shows how Model I and Model II operate on quite noisy shapes by removing irrelevant high frequency variations. Results are presented for the BUNNY (artificially corrupted with Gaussian noise) and the HELICOPTER shapes. Model I seems to work better on the BUNNY shape, while it tends to oversmooth the HELICOPTER shape. The loss of relevant information can be evaluated for the BUNNY shape by computing the L2 distance between the denoised shape and the original smooth shape. These error rates stay very low and similar to those observed in Table 3.1. Another observation however needs to be made. The very few points where the error gets larger correspond to high curvature regions that are very sensitive to noise corruption. In this case, it is almost impossible to recover the perfect noiseless signal without further knowledge.

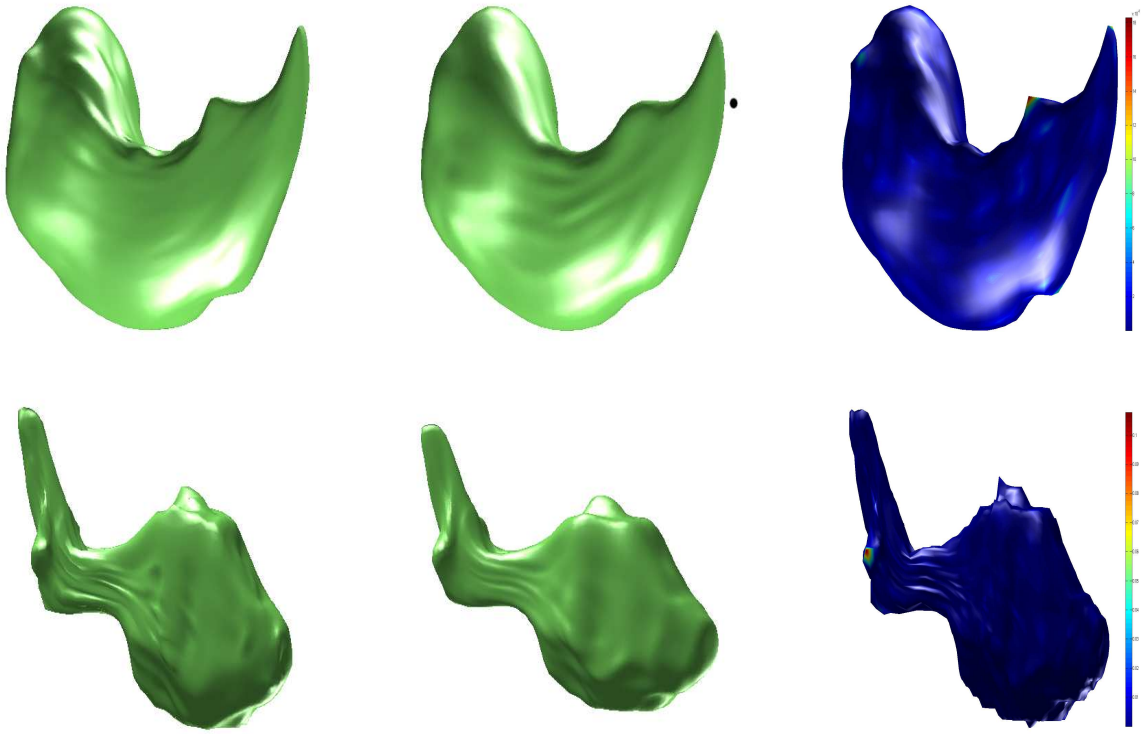
The results may be compared to those obtained with the double Laplacian smoothing technique that was proposed in [96]. Although the quality of this method depends



(a) (b) (c)

Figure 3.17: Smoothing of noisy shapes. First row: Results for the BUNNY, Second row: Results for the HELICOPTER. (a): Original shape, (b): Smoothed Shape with Model I, (c): Smoothed Shape with Model II.

upon the choice of two scale factors λ and μ, it is broadly used within the graphics community, works fast, prevents volume shrinkage and provides very satisfying results. The comparison of both methods is presented in Figure 3.18 for the HELICOPTER and HIPPOCAMPUS shapes. We can first observe that the smoothing obtained with the proposed technique is relatively close to that of Taubin. The mean distance between the two results is around 0.07% of the bounding box for the HIPPOCAMPUS shape, and barely reaches 0.2% for the HELICOPTER. However, one should note that smoothing seems to be slightly stronger with the Laplacian technique and more shape details are kept with the proposed method.



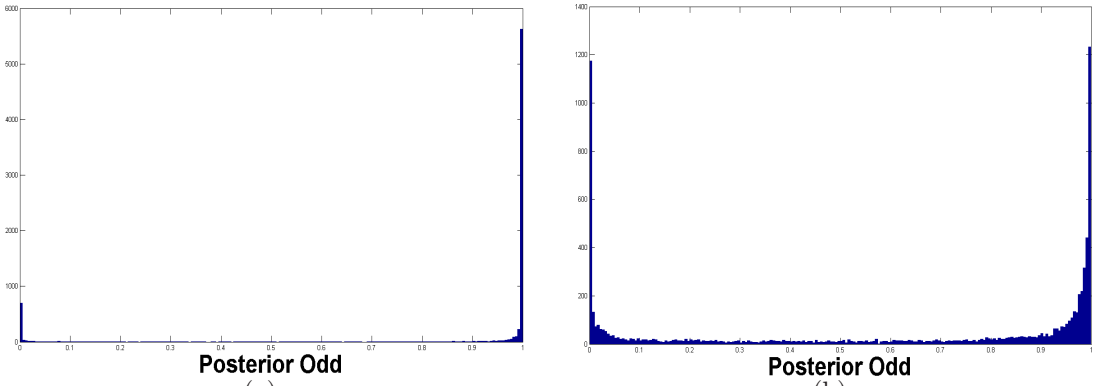
(a) (b) (c)

Figure 3.18: Comparison of smoothing between the proposed method and Taubin's Laplacian smoothing method. First row: HIPPOCAMPUS shape; Second row: HELICOPTER shape. (a): Proposed smoothing of the original shape, (b): Result of applying 40 steps of Taubin's smoothing method, with parameters λ=0.3 and μ=-0.33, (c): L2 distance between the two results.

3.6.2.3 Comparison of the Two Models

From the analysis of the experimental results for the two proposed methods, the following conclusions may be drawn:

- Model II yields lower reconstruction error (Table 3.1) and better smoothing with shapes that exhibit lower noise levels.
- Model I tends to offer higher compression rates (Table 3.1), while strengthening the smoothing rule with very noisy surfaces.
- Both methods provide good smoothing and compression results. The first one may serve as a validation tool to the second and vice versa.



(a) (b)

Figure 3.19: Distribution of the posterior odd values at the 6th level of resolution for (a): Model I, (b): Model II.

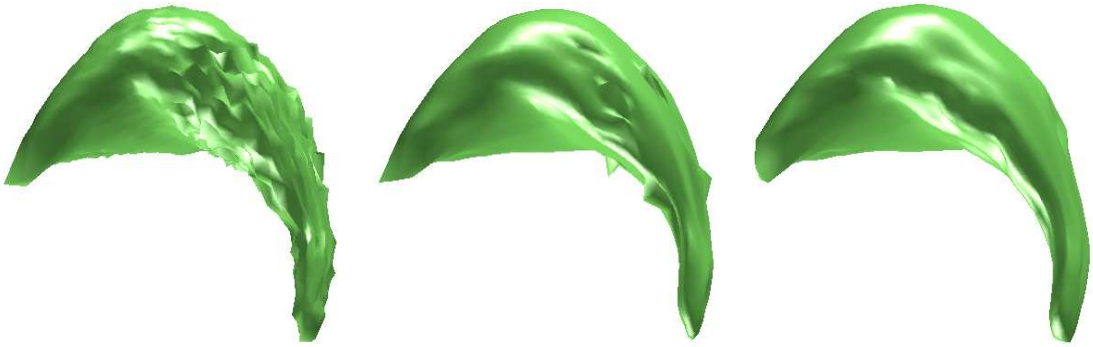
- The hyper-prior that describes the distribution of the noise level in Model II plays the role of an "extra layer" of Bayesian inference, and modifies the profile of the shrinkage rule. Figure 3.19 shows how the distribution of the posterior odd values changes with the choice of the method. Model I seems to classify wavelet coefficients in a more drastic fashion, while Model II presents more intermediate values.

For the remaining of this section, either one of the two methods will be used.

3.6.3 Robustness and Comparison to Other Shrinkage Techniques

It is reasonable to compare the proposed methods to classical wavelet shrinkage, for which an optimal threshold needs to be estimated for each resolution level. Several threshold level rules have been developed for first generation wavelets (see Section 1).

According to experimental observations, these rules were often numerically non-adapted to spherical wavelet shrinkage. Thus, universal thresholding [24], presented in Equation (3.23), seems to work decently with K=2 for classical wavelets, whereas better results are observed for K>6 in the spherical wavelet framework. However, no single universal value of K appears to work for all 3D meshes. A particular value K will provide very diverse smoothing for different shapes and different mesh sizes.



(a) (b) (c)

Figure 3.20: (a): Caudate shape affected by non-identical noise. The right part of the shape is corrupted by a much higher variance noise, (b): De-noised CAUDATE shape, using universal thresholding with K=9, (c): De-noised CAUDATE shape, using proposed thresholding.

Therefore, in order to be fair in the comparison of the proposed model with level-dependent universal thresholding, K is chosen such that the compression rate gets similar to that obtained with the proposed method. Thus, one may use K=9 for the CAUDATE, K=6 for the other shapes.

Another issue that is often encountered with such non-adaptive classical shrinkage methods appears when the shape is non uniformly affected by the noise. In order to test this and show how the proposed method handles this problem, different parts of the shape have been corrupted with different noise (i.e. with different variances) and observed the thresholding results. Figure 3.20 shows how the second shrinkage algorithm (Model II) outperforms non-adaptive classical shrinkage on the de-noising of the CAUDATE. The classical threshold is set up such that the number of coefficients that are shrunk is similar to that of the proposed method. Even then, classical smoothing is not able to consistently remove noise over the entire shape, whereas the proposed Bayesian method smooths the signal in a much adaptive manner. Local information allows the global Bayesian framework to gain in flexibility and operate in a data-driven fashion.

In Model I, the noise covariance matrix Σ is estimated at each resolution level.

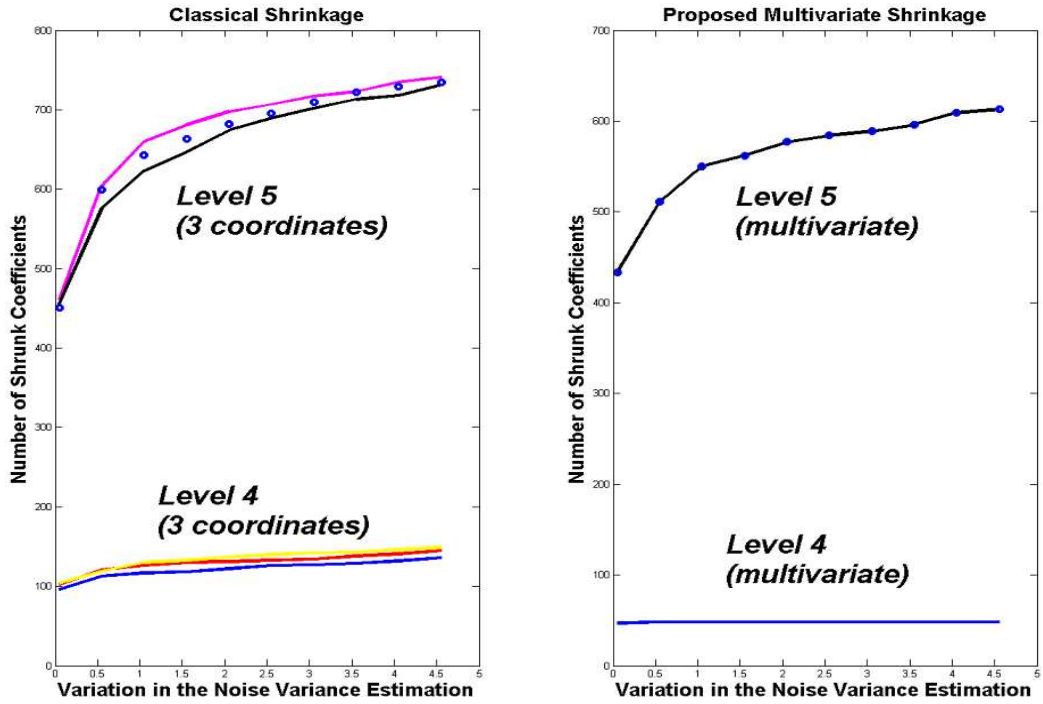


Figure 3.21: Sensitivity analysis (HELICOPTER shape): impact of the noise variance estimation on compression rates at two resolution levels. Left graph: universal thresholding, right graph: proposed thresholding

Different techniques may be used for noise variance estimation, but it remains difficult to rely on one specific method for all types of signals. These techniques often tend to overestimate noise power in some cases and under-estimate it in others. For these reasons, thresholding techniques should be robust to the errors in the noise variance estimation. In order to see the benefits of the proposed method on this specific point, sensitivity analysis has been run. Given a noisy shape, one can observe how thresholding results change as Σ varies. The proposed model has been compared to a non-adaptive universal model. In Figure 3.21, the evolution of compression rate with respect to the variations in the noise variance value is shown. The compression rate of the proposed method appears to be less sensitive to these variations, as it seems to "absorb" estimation errors.

3.6.4 Influence of the Curvature Term κ

It is interesting to see how parent coefficients and curvature bring in complementary information and improvement to the model.

The prior mixture weight π_0 carries crucial information in the proposed Bayesian framework. It can be regarded as a parameter of the model, which values vary around a mean value according to local information. If a constant value (π_0=Ct) were used instead, not only the choice of K would become primordial but the smoothing would also appear to be less coherent. Regions would be over-smoothed while others would keep irrelevant artifacts. That is what we observed as we replaced the adaptive π_0 with a constant $\bar{\pi_0}$=Ct into the prior distribution model and tested it on several shapes. The compression rate turned out to slightly increase (1%) but the final smoothing obviously exhibited a lack of coherence.

Furthermore, one should discuss the importance of the curvature term, when parent neighbors already bring some relevant local information on the environment surrounding a given coefficient. Indeed, since the parent coefficients of a given coefficient are defined at neighboring vertices, the curvature term $\bar{\kappa}$, which is also defined locally, may seem to bring redundant and unjustified supplementary information into the model. However, relevant curvature information is very often carried by more neighboring vertices than just those at which the direct parent coefficients are defined. We can show the importance of the curvature term by comparing the proposed model to a model with no curvature information (i.e. $\bar{\kappa}$=1). Table 3.2 illustrates this comparison by measuring the different rates of compression. We can see that the improvement brought by the curvature is real. However, we should note that the amplitude of the improvement varies with the type of shape. In the case of the HIPPOCAMPUS shape, large regions of low curvature exist and the additional compression when integrating $\bar{\kappa}$ in our model reaches almost 11%. An opposite example is the BUNNY shape, for which the surface does not exhibit wide flat regions. The

Table 3.2: Influence of Curvature

	No Curvature (κ=1)	Proposed Model	Additional Compression
BUNNY	38977	39731	+ 2%
CAUDATE	2301	2388	+4%
HELICOPTER	3581	3693	+3%
HIPPOCAMPUS	9001	9999	+11%

improvement in compression is therefore much lower (<2%).

3.7 Concluding Remarks and Discussion on Non-Spherical Surfaces

Experimental results show that the proposed wavelet shrinkage models work efficiently for the de-noising and compression of spherical surfaces.

As presented in Chapter II, wavelet decomposition of a signal defined on a non-spherical manifold can be performed using second generation wavelets. Thus, since the nature of the wavelet decomposition proposed in that framework is similar to that of the wavelet encoding scheme used in this chapter, it seems natural to broaden the application of the proposed wavelet shrinkage model to the class of non-spherical surfaces. Moreover, as mentioned in Chapter II, genus-one surfaces, because of area distortions introduced during the re-meshing process, may sometimes require very fine mesh structures in order to capture all the details of the shape. These meshes may therefore end up being over-sampled and generate a lot of low-valued wavelet coefficients. In this case, applying a localized wavelet shrinkage rule to the set of encoding coefficients should allow us to efficiently and significantly compress the signal.

CHAPTER IV

CLUSTERING METHODOLOGY FOR STUDYING RNA CONFORMATIONS

RiboNucleic Acid (RNA) is a large molecule (or polymer) consisting of a chain of *nucleotide* units that is involved in a multitude of biological activities in the cell [15]. Different types of RNA exist (rRNA, mRNA, ...), but its major role is to participate in protein synthesis.

Each RNA nucleotide consists of a nitrogenous base, a ribose sugar, and a phosphate. The phosphate and ribose sugar of the nucleotides constitute the *skeleton* of the RNA strand. This skeleton is usually referred to as the RNA *backbone*. The nitrogenous bases are attached to this backbone all along the molecule (see Figure 4.22).

The interpretation of the form of single stranded RNA molecules can be done at different levels, depending on the *structure* that is considered. Three types of structure are defined for RNA :

- The **primary structure** refers to the sequence of nucleotides. The primary structure can be easily assessed as it only refers to the nature of the constituents of an RNA branch.

- The **secondary structure** of an RNA molecule refers to the base-pairing interactions within a single RNA molecule. The secondary structure of an RNA can be represented in a plan and is uniquely decomposed into stems and loops. Much of the final structure is determined by the secondary structure and most of the secondary structure interactions consist of classical base pairing caused by hydrogen bonding. Other types of spatial conformations and 3D arrangements are not explained by this structure.

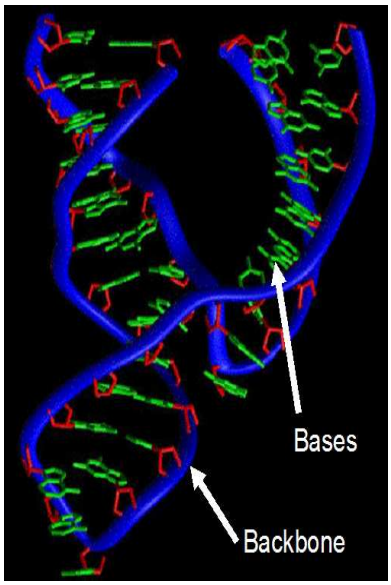
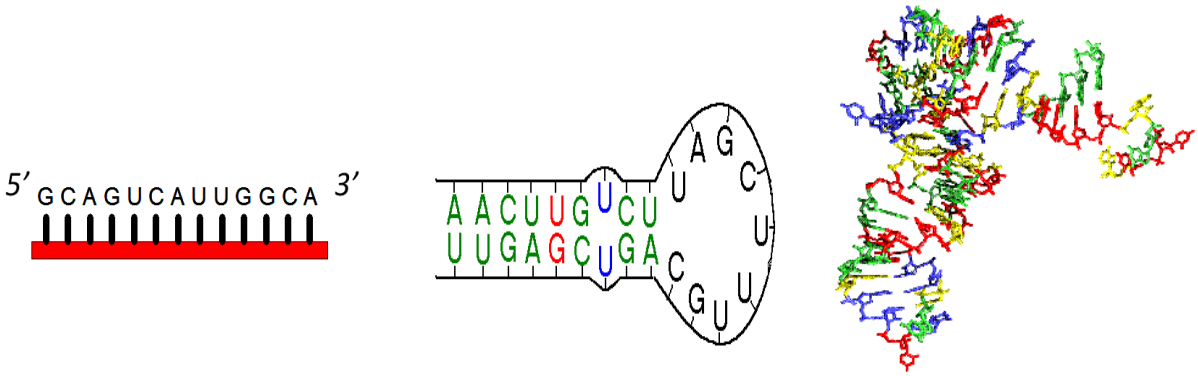


Figure 4.22: RNA strand composition: a sequence of nucleotide, where each nucleotide contains a "backbone" part and a base.

- The **tertiary structure** of an RNA molecule refers to the precise three-dimensional structure of a single RNA molecule. Major interest is shown by researchers for the analysis this structure as neither the nature of all existing types of interactions nor their exact relationship to the molecule functionality has been fully determined yet. This structure possesses a high degree of structural and functional variability.

The functional diversity of the RNA molecule depends on the ability of the RNA polymer to fold into a large number of precisely defined spatial forms (tertiary structure). Over the last few years, the data bank of RNA structures has considerably grown, due to major efforts made by experimentalists, leading to great advances in crystal growth protocols. The Nucleic Acid Database (NDB) [8] and the Protein Data Bank [9] accommodate the structural information of numerous RNA molecules, from small RNA structures with only a few nucleotides to very large ones with thousands of nucleotides, e.g., ribosomal RNA. The structural resolution achieved in these RNA molecules (obtained by the use of various characterization techniques) allows one to



(a) (b) (c)

Figure 4.23: RNA structure. (a): Primary structure: sequence of nucleotides, (b): Secondary structure: planar representation made of double helices and unpaired regions (loops) (c): Tertiary structure: Relevant level of organization for biological function (3D representation).

give an estimate for the location of individual atoms [76].

One of the main challenges of bioinformatics is to develop data mining tools for such documented RNA structures, in order to establish a clearer understanding of the structure/function relationships in these molecules. In most cases, this problem is too complicated to be solved computationally [91]. To date, efforts in this respect have focused on finding repetitive smaller substructures, i.e., *structural motifs* [71]. If the functionality of a specific substructure from a given structural motif is known, then the functionality of other substructures with a similar 3D form can be assumed to be similar. Therefore, the main task in the classification of these structural motifs is to define a similarity measure for substructures and to *cluster* motifs accordingly [41, 42, 73, 80].

The clustering problem is challenging due to several reasons: (a) the dimensionality of the data space may be high; (b) in most cases, prior knowledge of the number and/or structure of clusters is incomplete; (c) the chemical interactions between constituents are weak so that the boundaries between clusters are less pronounced; (d) in most cases, data density may be very low; and finally (e) the resolution of structures is low, compared to the needed atomistic level of accuracy.

This work aims to develop an efficient model for clustering the most basic building units of the RNA, namely, the single nucleotide and the nucleotide doublets resulting from interactions between bases.

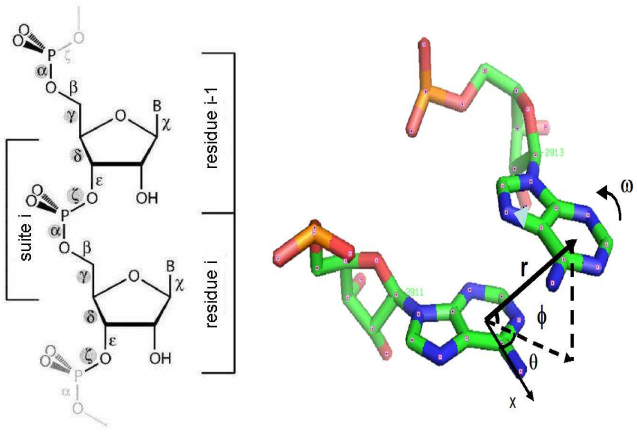
As mentioned above, RNA nucleotides are comprised of two distinct moieties: a flexible backbone consisting of ribose rings bridged by phosphate groups, and rigid bases consisting of either purines or pyrimidines. Most of the nucleotide interactions in an RNA molecule are due to interactions between bases. Given the differences between the flexible backbone and the rigid bases in RNA residues, the 3D structure can be described by two complementary representations (Figure 1): the backbone conformations [26, 27, 34, 41, 42, 85] of a single residue and the geometries of the base interactions [63].

The building block for the backbone consists of either the residue, or the base-to-base suite [73] (Figure 4.24-a). In the representation of the flexible backbone, residues are well-described by a set of six torsional angles, whereas suites necessitate considering seven torsional angles.

The representation of base interactions depends on six parameters, which describe the relative translation and rotation that are needed to align one base with the other. In this type of conformation, the coordinate system is composed of 3 rotation angles and a 3D vector representing the base-to-base distance. Note that the representation is not unique and depends on the choice of origin for the transformations.

Whereas the distances and angles are continuous parameters, differentiation of substructures and structural classification in both representations requires discrete criteria. For example, base pair geometries may be organized into twelve classes with respect to the interacting edges of the bases [62]. Single nucleotide conformations can be classified into groups of rotamers [73].

For both representations, the recognition and definition of the classes are formulated as a segmentation problem, which deals with partitioning of the continuous



(a) (b)

Figure 4.24: (a) RNA backbone with six torsion angles labeled on the central bond of the four atoms defining each dihedral. The two alternative ways of parsing out a repeat are indicated: A traditional nucleotide residue goes from phosphate to phosphate, whereas an RNA suite, which is more appropriate for local geometry analysis, goes from sugar to sugar (or base to base). (b) A base base interaction with a possible parametrization.

data space into a finite collection of well-defined subspaces. This segmentation is done by recognizing the underlying clusters in the data space. More precisely, each data point will correspond to an instantiation of the conformation under investigation (backbone conformation, base-to-base interaction geometry) and will be defined in a d-dimensional space, where d is the number of parameters that fully defines the spatial representation of the conformation.

The work presented in this chapter proposes applying a clustering algorithm based on the Potts model to the problem of data mining of RNA structures. The remainder of this chapter is organized as follows. In the next section, a brief overview of existing clustering methods and their applications in bioinformatics is given. Next, Sections 4.2 and 4.3 describe and discuss the Potts model that has been employed for clustering. In Sections 4.4 and 4.5, specific applications to the single and double nucleotide classification problems are presented, and the results are compared to some known classification. Then, the results of the clustering method for the base stacking problem

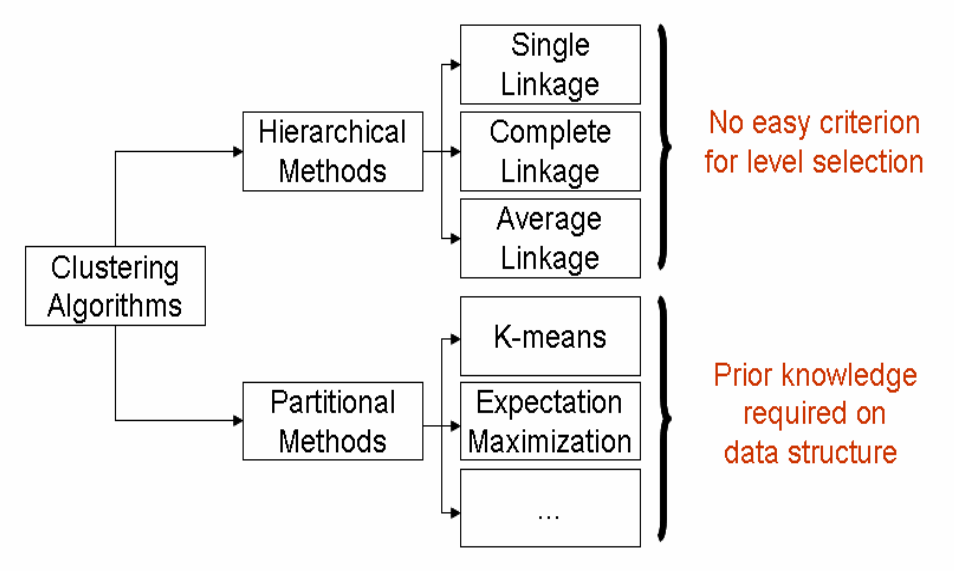


Figure 4.25: Classification of clustering methods.

are reported. Finally, in Section 4.6, conclusions are drawn and remarks on the proposed classification are made.

4.1 Overview of Clustering Techniques and Motivation for a Non-Parametric Model

Attempts to perform structural data mining on RNA have been done using either qualitative observations of the data space [42, 80]. This "manual" classification technique turns out to be tedious and inaccurate since the dimensionality of the data space can be larger than three, and because the distribution of data points can be fuzzy. Thus, the need for an automated clustering methodology is real.

Data clustering consists of assigning a set of observations into subsets (called clusters) so that observations in the same cluster are similar, with respect to a given distance measure. There are numerous different clustering methods. Yet, the most common algorithms can be classified into two major categories: *Hierarchical* methods and *partitional* methods (see Figure 4.25).

Hierarchical algorithms [32, 47] build successive clusters using previously established clusters. These algorithms are either *agglomerative* or *divisive*. Agglomerative

algorithms begin with one data point as a separate cluster and successively incorporate other points into the existing clusters to form new larger clusters. Divisive algorithms begin with the whole set as a cluster and successively divide it into smaller clusters. Hence, intermediate clustering configurations obtained through this iterative decomposition can be represented graphically by a solution tree, usually referred as *dendrogram*. In this tree, each branch splitting corresponds to the merging (or splitting) of two clusters. In hierarchical methods, the construction of the clusters does not only depend on the metric used to calculate distances, but it also varies with the nature of the *linkage* criterion. This criterion defines the way distances between clusters are calculated and the order in which clusters are successively formed, split and merged. *Single linkage* (respectively, *complete linkage*) clustering uses the minimal (respectively, maximal) pairwise distance that exists between the two clusters. *Average linkage* clustering computes the average distance between two points from the two clusters. The choice of the linkage criteria may depend on the structure of the data set, on the objective of the clustering algorithm or on any other external factor defined by the end-user. Hierarchical methods are commonly used for various types of data classification. However, not only the choice of the linkage criteria, but also the depth at which one needs to look into the dendrogram, are two major drawbacks of these algorithms. Indeed, in many cases, neither the number of clusters nor the structure and distribution of the data is known. Since no such information is provided for the clustering of RNA conformations, previous attempts to perform clustering using hierarchical methods did not offer a very well-defined framework [60].

Partitional algorithms typically determine all clusters at once. In this category, two major techniques are commonly used in bioinformatics. K-means clustering consists in assigning a data point to the cluster whose *centroid* is nearest. The centroid is computed to be the average spatial location of all the points in the cluster. This

method has been used for studying RNA conformations [41, 94], but the main drawback of this algorithm remains that an assumption needs to be made on the number of clusters beforehand. The K-means algorithm is therefore referred to as a *parametric* clustering method. Imposing a pre-defined number of clusters would therefore prevent the user from finding new relevant RNA sub-structure configurations. Another well-known partitional algorithm that has been used in bionformatics [61] is the expectation-maximization algorithm (EM) [21]. This method assumes that the distribution of the data points from a given cluster is Gaussian and aims to find the optimal parameters of the corresponding mixture of Gaussian distributions. This iterative algorithm consists of a succession of "expectation" and "maximization" steps. Initially, the model parameters (i.e. the center and the spread of each cluster, as well as the probability for each data point to belong to one of these clusters) are estimated. During the "expectation" step, the corresponding multi-Gaussian likelihood is computed, while, in the next "maximization" step, new parameters are estimated by maximizing that quantity. This process is repeated until it converges. Obviously, the major drawback of this method is the assumption made on the data distribution. In the case of RNA clustering, the underlying distribution of the data points is sometimes complex, due to the poor resolution, and the data structure can certainly not be approximated by a mixture of Gaussians. Thus, this type of clustering is not well-adapted to the work presented in this thesis.

Even though some of these classical clustering methods have provided satisfactory results for the classification of RNA confirmations, the ideal model would consist of a non-parametric and fully automated algorithm that makes no assumption on the structure of the data or the number of clusters.

Among the most promising alternatives for the clustering of such data is a method based on the statistical-mechanical Potts model [10]. In this approach, a spin parameter is assigned to each data point, and an interaction parameter is attached to each

pair of neighboring data points. The closer two points are, the stronger the interaction between them, and the more likely they will belong to the same cluster. The partition of the data space into clusters can be found by employing spin-spin correlation functions. The merit of this technique is that no prior knowledge of the data point distribution is needed in order to devise the different parameters. Indeed, the Potts model is a non-parametric hierarchical clustering approach which gives intrinsic and objective criteria for defining the correct level of the hierarchy for producing clusters that optimally match the underlying physical model. This is done by linking the data space to the space of an alternative physical problem where one tries to find paramagnetic regions in a potential induced arrangement of magnetic particles. *Susceptibility* and *temperature* turn out to be natural parameters and variables for describing the clustering problem. *Temperature* plays the role of hierarchy level or *depth* in the iterative subdivision of the data into clusters. The *susceptibility* graph gives a simple criterion for selecting the temperature at which clusters are determined [10]. The values of other parameters intrinsic to the Potts model can be optimized by the use of some straightforward numerical criteria [4, 10]. The Potts model is advantageous relative to standard parametric methods by virtue of the fact that no prior knowledge about the number of clusters is needed, and is also much simpler to employ than most other non-parametric algorithms.

In the next section, a brief description of the Potts model will help the reader get familiar with the terminology.

4.2 Background on Data Clustering using the Potts Model

As *a priori* information on the number and the size of conformational classes may not be available for a given data set, a non-parametric clustering method fits our RNA structure classification problem. Plus, such methods are more suitable to find new elements in the classification. The method that is presented here has been proposed

by Blatt *et al.* [10] and is based on a Potts spin model, which was developed for the analogous problem of the physical properties of an inhomogeneous ferromagnet. In this model, clusters consist of magnetic islands containing sites with similar Potts states.

4.2.1 Description of the model

We now give some specific details on the Potts model. We refer to the N points of the given data set as *magnetic sites*. Each site is assigned a Potts spin denoted by s. Spin values are taken from a set of q distinct integers, where q is a parameter to be set. The allocation of a spin value to every magnetic site results in a unique spin configuration S that entirely defines the state of the system. One can then define q^N different spin configurations. Moreover, the spins s_i and s_j of two sites i and j are said to be "aligned" if they have the same value (i.e. if $s_i = s_j$).

Also, two spins s_i and s_j interact with each other with a strength J_{ij}. For computational reasons one assumes that a given spin will have a significant interaction with some of its closest neighbors only [10]. The method for choosing neighbors is described in Section 4.3.1.

For any given spin configuration S, the energy of the system is defined by the following Hamiltonian:

$$H(S) = \sum_{<i,j>} J_{ij} \cdot (1 - \delta_{s_i s_j}), \tag{4.58}$$

where $\delta_{s_i s_j} = 1$ if $s_i = s_j$ and $\delta_{s_i s_j} = 0$ otherwise. J_{ij} is the interaction between two neighboring sites i, j. Here we choose $J_{ij} = \frac{1}{K} \exp(-\frac{d_{ij}^2}{2a^2})$, where d_{ij} is the Euclidean distance between i and j, a is a normalization constant, and $\overline{K}$ the average number of neighbors for a given site. We assume that no interaction exists between non-neighboring sites.

Note that, according to the energy in Equation (4.58), two sites with high mutual interaction will pay a "high energy price" if they are not aligned.

The probability to find the system in a given spin configuration S depends upon a *temperature* parameter T, and accordingly, one defines the probability density P_T:

$$P_T(S) = \frac{1}{Z} \exp(-\frac{H(S)}{T}), \tag{4.59}$$

where Z is a normalization constant.

For a given temperature, one can compute the thermodynamic average of any quantity by estimating the weighted-average of this quantity over all possible spin configurations with respect to $P_T(S)$.

4.2.2 Key quantities and metrics for clustering

Within this framework, we now describe the details of the clustering process.

4.2.2.1 Spin-spin Correlation and Clustering

At a given temperature T, clusters are formed by grouping sites that are most likely aligned, with respect to the corresponding probability distribution P_T. The key element in this clustering process is the introduction of the spin-spin correlation G_{ij} defined between two sites i and j, which represents the probability for two spins s_i and s_j to be aligned. Two neighboring sites i and j are most likely aligned if their spin-spin correlation value G_{ij} is high (typically greater than 0.5). In such case, a link is set between these two sites, and then the two sites are taken to belong to the same cluster. By applying this rule to all pairs of neighboring sites one can thus easily build connected graphs. A cluster, referred to as a *magnetic grain* in the Potts model, is then defined as one of these connected graphs.

4.2.2.2 Order Parameter and Thermodynamic Phases

Clusters evolve with temperature. At lower temperatures, larger clusters are formed, whereas higher temperatures allow for more disorganization and less clustering. Some of the temperatures will correspond to major structural changes in the cluster organization, and will delimit specific thermodynamic phases. In order to concretely exhibit

these different phases, one needs to consider the average magnetization of the system, $\langle m \rangle_T$, which measures the degree of ordering of the system at each temperature. Details on the computation of the average magnetization are described in [10]. By considering the variations of the degree of ordering as the temperature changes, one can distinguish among different thermodynamic phases. At low temperatures, the *ferromagnetic* phase is characterized by a well-ordered system and the presence of only one major magnetic cluster. As the temperature increases, we move to the *superparamagnetic* phase, in which clusters successively break down into distinct magnetic grains. Finally, at very high temperatures, the system gets totally disordered; this is the *paramagnetic* phase.

Transitions between phases can be investigated by evaluating the susceptibility χ, which tells us about the variance of the magnetization:

$$\chi = \frac{N}{T}(\langle m^2 \rangle_T - \langle m \rangle_T^2). \tag{4.60}$$

Large fluctuations in the susceptibility characterize successive subdivisions of the magnetic grains. Hence, one wants to detect major peaks in the susceptibility and therefore determine at which temperatures major subdivisions occur in the cluster decomposition process. Each major peak corresponds to either cluster splitting or cluster disaggregation (i.e., the cluster melts away). Thus the system defines a metastable state configuration over an interval of temperatures that is delimited by two successive peaks. However, as we observe the configuration of the system over such an interval of temperatures, we notice that some of the data points (usually located at the fringe of their clusters) will tend to successively dissociate from the clusters as temperature increases, without creating any major effect on the whole clustering configuration. Accordingly, the temperatures that immediately follow a peak in the susceptibility graph are those temperatures at which we will perform and analyze clustering. More specifically, temperatures that are chosen for clustering will be taken at local minima that immediately follow a peak in the susceptibility graph. Finally,

at higher temperatures, an abrupt decrease in χ characterizes the transition to the paramagnetic phase.

4.2.3 Monte Carlo Simulation

The computations of the average magnetization and the spin-spin correlation both involve the notion of thermodynamic averaging. As proposed in [10], a Monte Carlo simulation allows us to generate M configuration samples at each temperature, with respect to $P_T(S)$. The method is based on a Markov chain process generated by the implementation of the Swendsen-Wang Monte Carlo algorithm [101]. This Monte Carlo approach turns out to be computationally efficient, by enabling us to flip a whole set of spins in one iteration, instead of changing the configuration one site at a time.

4.3 Discussion on the Method

Even though this Potts model-based clustering is considered to be a non-parametric algorithm, nevertheless many parameters need to be adjusted. Those parameters usually allow for flexibility in the process of forming clusters. We now detail the analysis of the most significant parameters.

4.3.1 Mutual Neighbors

As previously mentioned, non-neighboring sites have no interaction. Several different options exist to define the concept of neighborhood [10]. Here, we characterize neighbors using standard mutual neighborhood conditions. Two sites are *mutual neighbors with value* K if each is a K-nearest neighbor of the other.

The chosen value K can vary from 0 to N-1, and this choice will have a potential impact on the outcome of the algorithm. Instead of employing a homogeneity parameter to find K as in [4], we propose a novel approach using coordination numbers. Indeed, for our data sets, the type of method described in [4] turns out to provide very

large values of K. However, very large values of K not only make the algorithm computationally expensive but also tend to shrink the thermodynamic region of interest for clustering. We therefore define and use another criterion, based on sphere packing theory, in order to efficiently choose the parameter K. Sphere packing theories study the arrangements of spheres in a given volume, and the resulting densities. In this manner, they naturally connect to the notion of coordination number.

Specifically, in this work, we model our sites as equal-radius spheres, and assume a maximal density. The number of nearest neighbors for this case is also known as the *kissing number problem*. A kissing number is the maximal number of spheres that can touch a single sphere in d dimensions. In d-dimensions for $d = 2$ or 3 [30], any given sphere has $6(d-1)$ closest neighbors, and so for any given site, we pick its $6(d-1)$ closest neighbors, and consider only these as potential mutual neighbors. This method allows us to have a standard and straightforward way to choose K, while providing quite reasonable results.

4.3.2 Advantages of the Potts Model and Comparison to Classical Methods

The Potts model clustering carries advantages over other non-parametric clustering algorithms, such as hierarchical linkage methods [32,47]. As mentioned in Section 4.1, these linkage clustering methodologies present several undesirable characteristics. In this section, two points are discussed that highlight the major advantages of the Potts model over these hierarchical methods.

First, no clear and unambiguous indication is given about the depth at which one should explore the tree in order to obtain reasonable clustering. The major steps of the iterative cluster construction are usually not easily detectable. In [14], a criterion is defined that tries to provide a way to find appropriate levels in the decomposition at which clustering should be performed. However, this criterion is not necessarily straightforward to use. Therefore, even with such a criterion, the optimal clustering

configuration becomes more difficult to estimate for a non-expert analyst, since a large number of potential configurations are available along the tree. The Potts model, reduces the amount of uncertainty by limiting the decision making process. Indeed, very few peaks are usually observed over the superparamagnetic phase, and one only needs to choose among those few peaks to obtain a final clustering.

Second, an issue of concern with hierarchical linkage methods is the legitimacy of clustering in certain cases. Indeed, when no real inhomogeneity exists in the data set, clustering should not occur. However, with classical methods, the decomposition nevertheless proceeds and generates a non-trivial tree. When the data set is homogeneous, the Potts model does not generate any intermediate clustering, since no superparamagnetic phase is observed, i.e., the configuration jumps from one large cluster to no cluster without any transition. To illustrate this, we applied the Potts model clustering to points sets with no real inhomogeneity in the distribution. Thus, we generated ten data sets of 2-D random variables, uniformly distributed over $[0; 1]^2$, and observed the susceptibility graphs for each of these (Figure 4.26). Note that each of these graphs exhibits only one peak and that none of them cases shows a plateau after the peak, meaning that no superparamagnetic phase exists, and no clustering configuration is formed. Not so, when we applied an hierarchical clustering method on the same random sets we always got clear clusters.

The next sections describe the results of applying the Potts clustering method to the RNA structure problem. The proposed work is aimed to show how this clustering algorithm can efficiently lead to a classification of RNA conformations and sub-structures. Concretely, each point from the input data corresponds to an instantiation of the conformation under investigation (backbone conformation, base-to-base interaction geometry) and will be defined in a d-dimensional space, where d is the number of parameters that fully describes the spatial representation of the conformation.

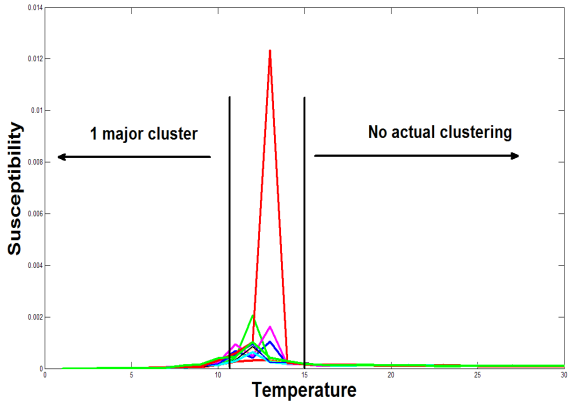


Figure 4.26: Susceptibility graphs for 10 different random generated data sets. No real clustering phase is observed.

First, the Potts model is used to validate existing classifications (for backbone conformations and base-pair geometries). Second, it is applied to the problem of finding clustering for the base stacking case, which to the best of our knowledge, has not been solved in the past.

4.4 Backbone Structural Conformation Classification

The backbone structural conformations of RNA can be represented by either residues or suites. The structural flexibility of a residue (or suite) stems from the modes of motion of its backbone. Potential modes of motion for nucleotide backbones are restricted to N_{Tor} rotations around covalent bonds where $N_{Tor} = 6$ for a residue or $N_{Tor} = 7$) for a suite. Accordingly, we describe the single nucleotide conformation using N_{Tor} angle parameters (Figure 4.24) Rotations of the backbone are restricted by molecular forces. Due to these restrictions, the backbone conformation distribution in the N_{Tor} dimensional torsion space is strongly non-homogeneous. The data points are mostly restricted to lie in clusters consisting of disconnected regions of the torsional space. This clustering characteristic can be used as a similarity criterion for classification of conformations for a single residue or suite. Two conformations are considered to be similar only if they reside within the same cluster. Such clustering

was performed via qualitative observations using projections of the data space onto sub-dimensional spaces. Thus, for residues [42], a representation of the data in six separate histograms was proposed in order to analyze the six torsional angles of the residue conformation. For suites [73], 3-dimensional projections of the torsional space showed an alternative for dimensional decomposition. In both cases, classification is made difficult by the high dimensionality of the data space.

The non-homogeneous distribution of data points makes the problem a good candidate for the application of an automated clustering method to the original data space. In [42], a k-means based algorithm was used to find clusters in the single nucleotide (residue) conformation. The efficiency of that method is hindered by several factors: (a) there is no prior knowledge on the number of clusters; (b) k-means is based on the definition of a global metric, while actual physical forces in the RNA structure dictate a local metric that is unknown. The Potts algorithm seems to be a good candidate to find clusters in the conformation data space, since it does not require any prior knowledge about distribution of clusters and since it is based only on a nearest-neighbor criterion. We have performed a clustering analysis for both residue and suite representations.

4.4.1 Single Residue Cluster Analysis

The data that we used to examine the algorithm was composed of approximately 2800 single nucleotide conformations from the structure of the ribosome of HM LSU 23S [7] (RR0033 in NDB). This data test case was chosen in order to compare the clustering results with previous clustering scheme [42] . This structure has a high accuracy (resolution $2.4\mathring{A}$) and is often used as a test case for structural data mining of RNA. Clustering was performed on a four dimensional data space using the four discriminating torsional angles [41, 42] α, γ, δ and ζ. Also, as torsion angles α, γ and ζ ($\in [0; 360]$) are circular dimensions, adjustments have to be made when considering

distance, neighborhoods and interactions between points. Thus, for each dimension, our algorithm accounts for this cyclicity by allowing angles close to 360 degrees to be "neighbors" of those close to 0 degree. Interaction between sites is similarly adapted by choosing the shortest distance between two points among all the possibilities that exist when considering all trigonometric directions.

The parameters that we used in the algorithm were carefully chosen in order to optimize and facilitate the visualization of the results. We tested different values of q, the number of possible spins, and finally used q=20 for the particular application of the algorithm to backbone conformations, as well as for all other classifications that are presented in the remainder of this chapter. A distance normalization parameter a equal to the average distance between mutual neighbors was found to be satisfactory. No major difference was found when scaling that parameter (we tested it for both $2a$ and $a/2$). The number of nearest neighbors K was chosen to be 18, based on an extrapolation of the explanation in Section 4.3.1. The calculated susceptibility diagram for backbone conformation analysis is shown in Figure 4.27. Based on this graph, we can detect three main transitions in the evolution of the clusters. As previously mentioned, major transitions are represented by peaks in the susceptibility diagram. Over an interval of temperatures that is delimited by two successive peaks, we choose temperatures for clustering analysis as explained in Section 4.2.2.2: temperatures that are chosen for clustering will be taken at local minima that immediately follow a peak in the susceptibility graph. Thus, for the case of single residue conformations, we performed cluster analysis at three different temperatures. The first two temperatures were chosen to be $T_1 = 0.016$ and $T_2 = 0.061$. These two temperatures just followed the first two peaks that we observed in the susceptibility graph (see Figure 4.27). The third temperature, ($T_3 = 0.126$), represented the formation of a single new cluster, which corresponded to a typical conformation of a residue in the 3' side of an A-form RNA helix. In Figure 4.27, these three temperatures are marked

by three vertical lines.

The analysis of the clustering may not seem straightforward because of the multi-peak structure of the susceptibility diagram. To address this issue, we first performed a cluster analysis at each of the three aforementioned temperatures, from the lowest to the highest. If a cluster was found at a low temperature and survived as a single cluster at higher temperatures, then this cluster was taken into account in the classification. If, on the contrary, a cluster was found at a low temperature but was then split into two or more significant clusters, then the new clusters were kept for analysis. Note that there can be no relevant splitting after the third peak because the fourth peak represents the final melt-down of all clusters and the transition to the paramagnetic phase.

Also, comparison of our results to those from the binning method [42] allowed us to validate the pertinence of our classification. This validation was mostly limited to a decision on the "cutoff" size of the clusters (i.e., the minimal size above which a cluster is taken into consideration). Indeed, such information may be needed for the decision if a given cluster is melting or splitting

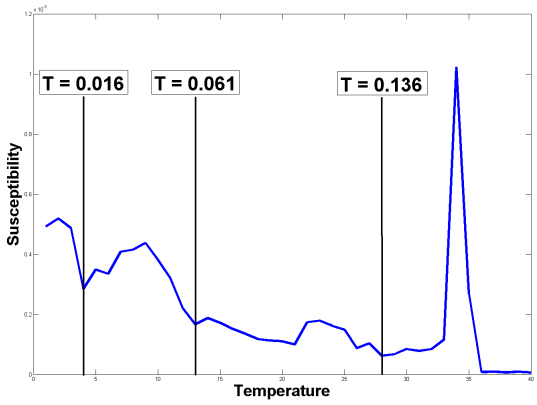


Figure 4.27: Susceptibility graph for residue conformation (RR0033). We observe the three successive peaks that are responsible for the major transitions in the cluster configuration. Temperatures for clustering are taken at the local minima that immediately follow each one of these peaks.

To validate our analysis, we have quantitatively compared the Potts and binning clustering techniques. The minimal size (or "cutoff" size) of a cluster was chosen to be six. The results of the clustering analysis, and the correspondence between the binning and Potts models, are presented in Table 4.3. The first column contains the bin index, using the alphabetical annotation as proposed in [42]. In the second column, each bin is allocated a four-digit number where each digit encodes the spatial range of each of the four torsional angles [42] (as detailed in Table 4.4): α, γ, δ and ζ. The third column gives the index of the peak after which the cluster was first identified. The fourth column contains the indices of the "corresponding" clusters found with the Potts model. The fifth column gives the number of residues that are found in both the bin and the corresponding Potts cluster as compared to the total number of residues in the bin. The sixth column compares the same number of common residues to the number of residues in the Potts cluster. The seventh column contains additional information of the typical functionalities of the residues within each cluster. As can be seen from Table 4.3, there is a very good agreement between the two methods, except for few cases. The first case consists of very small clusters that have split out of a larger bin (d and d', y and y', a and a') or very small bins that have not been recognized as clusters by the Potts method. The second and significant case consists of clusters e and a. According to the binning method, these two clusters include nucleotides that reside within the interior of the A-form RNA helical region (a) or the 3' end of it (e). The difference between these two conformations was in the angle ζ. For the bin a, ζ is in the g- conformation (Table 4.4) and for e it is in the g+ or t conformation. According to the Potts method, cluster a has the same range of angles, but cluster e includes only the g+ range of ζ. The residues with ζ in the t conformation have melted away without forming any clusters. It is important to note that the cases where residues in the e Potts clusters are those that take part in known motifs (E-loops and kink turns), while residues that are in bin e but not in

Table 4.3: Residue Conformation Classification: Correspondence Between Binning and Potts Clustering

Bin letter	Bin number	Peak	Potts cluster	Cluster/Bin	Bin/Cluster	Remarks
a	3111	third	a	1545/1766	1545/1545	A-form RNA helix
a	3111	second	a'	11	11	base on the 3' side bulge out
e	3112	third	e	40/160	40/40	3' end of an A-form RNA ζ is in g+
i	2211	second	i	105/113	105/107	crank shaft in A-form RNA
r	3122	second	r	104/133	104/104	interstrand stacking
d	1322	first	d	11/18	11/15	take part in kink-turn
d	1322	second	d'	6	6	ζ in t
c	1121	first	c	32/32	32/38	take part in kink-turn
n	3121	second	n	33/40	33/40	non contengeous stack both 3'5' directions
o	2111	second	o	61/68	61/71	the turn in Tetra loop
l	1211	first	l	38/38	38/39	interstrand stack or (i,i-2) stack
t	1111	second	t	38/41	38/43	stacked between the 5' adjacent residue to 3' non adjacent
u	3211	second	u	24/32	24/26	hinge between two helical strands
s	2122	first	s	34/38	34/34	take part in e-loop and kink-turn
h	3222	first	h	9/15	8/9	take part in e-loop
g,7	2121,4121	first	g	8/9	8/17	bulged residue, base often non stacked
v	3311	first	v	7/11	7/7	
m	1122	first	m	10/18	10/14	
f	1112	first	f	8/15	8/8	
3	1221	first	3	8/8	8/8	
y	1311	first	y	7/15	7/15	crank shaft in A-form RNA
y	1311	first	y'	6/15	6/7	
1	3321	first	1	8/9	8/10	
0	1222	first	0	4/5	4/6	
	3312	first	3312	5/5	5/6	

Table 4.4: Delimitation of the Bins in the Binning Method

α	γ	δ	ζ
40-100	10-110	65-105	240-350
125-200	140-210	130-165	other
220-350	230-350	other	
other	other		

cluster e do not have such affiliation.

4.4.2 Suite Cluster Analysis

As a second test case, we have performed cluster analysis to validate the suite structure as presented in Richardson's work [73]. First, we applied Potts clustering to the simplified case of a five-dimensional representation of the suites, where five identifier angles are taken into account: δ_{-1}, ζ_{-1}, α, γ, δ [73]. We used the RR0033 data base for this suite clustering analysis in the 5D conformational space and this method was shown to give a relatively good agreement with the full 7-dimensional suite torsional

space.

After validating the method on the simpler 5D case, we applied the clustering algorithm on the full seven-dimensional suite representation, where the seven identifier angles are: δ_{-1}, ϵ_{-1}, ζ_{-1}, α, β, γ, δ. For this case, we used the RNA05 data base as in Richardson's work [73]. This data base includes more than 9000 suites from different RNA structures. Given the size of the data set, building a fine susceptibility diagram was found to be very time consuming. In this context, we used coarser intervals of temperatures, making the detection of local minima more difficult. In order to validate our classification, we used the same algorithm as in the single residue case but with temperature increments of 0.002. We also added a stability criterion for each cluster that was found. According to this criterion, a cluster is said to be stable if it "survives" (i.e. does not undergo major modifications) for at least two adjacent temperatures. This criterion allows one to disregard the clusters that may form around the susceptibility peaks but that actually correspond to undesired, metastable states. The analysis reveals that the majority of the clusters have been formed at the first peak. The result of the cluster analysis is presented in Table 4.5. This table has been built in order to mimic Table 1 in [80]. The first column of the table gives the ASCII annotation of the bin corresponding to the given 7D cluster. When no annotation existed, we used the numerical annotation described in Section 4.4.1. The second column gives the total number of data points in the cluster. The third column gives the bin ASCII code of the corresponding 5D representation of the cluster [80]. The fourth column gives the number of data points that are both in the Potts cluster and its associated 5D cluster bin. The fifth column gives the consensus suite cluster [80] that agrees with the new 7D Potts cluster . If no suite cluster was found to agree with the new Potts cluster, we left the entry empty. The sixth column lists an example from the RNA05 data file of a suite with a conformation that that is typical of the cluster. To keep this column consistent with previous results, we used the same

example as in Table 1 in [80], whenever this was possible. The seven other columns give the mean dihedral angle values for each of the Potts clusters with the standard deviation in parenthesis. For most cases, there is a very good agreement between the consensus suite clusters, the 5D bin suite clusters and the new Potts clusters. The most significant difference between the new Potts clusters and the consensus suite clusters is the decrease in the number of clusters. Indeed, while the consensus suite clusters includes 46 clusters, the Potts clustering algorithm generates only 32 clusters. For most cases, this decrease results from the merging of some consensus suite clusters into a large Potts cluster. For example, consensus suite clusters $1a$, $1m$, $1L$, $7a$, $9a$ and $6g$ merge all into cluster a of the Potts classification. Either these clusters were not distinguishable from each other at any temperature, or they melted away from the main cluster without forming any relevant individual clusters. Some other small consensus suite clusters, such as $6j$, were not recognized as clusters by our classification. The Potts classification introduces several new clusters, among which some represent a "crankshaft" of the A-form RNA. This is cluster y where the the "crankshaft" effect is manifested by the transition α: $g_- \rightarrow g_+$ and γ_{-1}: $g_+ \rightarrow g_-$. The second "crankshaft" conformation is a variation of cluster i, where the transition is in the angle ϵ_{-1}: $g_- \rightarrow g_+$. Another new cluster is cluster u' that similarly to cluster u, includes a bend conformation in an A-form RNA single strand, but does not participate in an kink-turn motif. Cluster E is similar in conformation to cluster E', but the angle ζ_{-1} of the Potts cluster is restricted to the g_+ orientation. Cluster F' is within very close range to cluster F, but the angle ϵ_{-1} is shifted from the g_- to the trans orientation. The new cluster does not force the bulge that appears in the main cluster F [80]. One more cluster, the 12231 cluster, seems to be a stable cluster that does not appear in other classifications.

In summary, it seems that the new technique removes some splitting between clusters as compared to the consensus suite clustering [73], but also introduces several

new clusters. We should note that several of the clusters that were not present in the conformer library shown in [73] have an irregular ϵ_{-1} value. It is remarked in [73] that irregular ϵ_{-1} torsions are frequently found in RNA structures that have been improperly fit into electron density. Thus, it is likely that the clusters presented here represent certain common misfits that were intentionally excluded from the conformer library in [73]. The obvious advantage of the Potts clustering technique is that it is almost un-supervised. After finding several parameters such as the temperature range, the temperature interval of stability and the cutoff size of a cluster, the application of the technique becomes fully automated.

4.5 Base Doublet Geometry Classification

4.5.1 Coordinate Systems

Base doublet interactions may be of different types. Thus, in this work, we applied the proposed clustering method to both base pair and base stacking interactions. For both cases, we employed two parametrization methods, which we would like to detail here.

The first coordinate system was proposed in [84]. It is defined by considering only translation and rotation parameters, rather than relative hydrogen bond distances. More specifically, a set of three parameters was used. Two parameters defined the projection of the glycosidic $N1/N9$ (pyrimidine/purine) atom of one of the bases on the plane of the other base. The third parameter defined the rotation of one of the bases around its center of mass that was required to align it with the second base (see Figure 4.28-a). In the remainder of this text, we refer to this parametrization as the *center of mass* (COM) parametrization, and we use it as a reference case.

The alternative parametrization that has been employed in this work is based on the Pople coordinate system [19] with the origin in the center of the pyrimidine ring

Table 4.5: Suite Conformation Classification for RNA05 (7D): Comparison of Potts results with Previous Nomenclature

Cluster	Points in cluster	Bin Ascii	Points in bin	Cluster 2008 (Consensus)	Example	δ_{-1}	ϵ_{-1}	ζ_{-1}	α	β	γ	δ
a	5544	a	5544	1a,1m,1L,7a,9a,6g	UR0020-11	82(3)	211(10)	289(8)	296(8)	174(9)	53(6)	81(3)
a'	17	a	15	#a	RR0082-1940	82(5)	193(7)	256(7)	301(8)	174(6)	51(9)	80(7)
o	111	o	111	1g	RR0082-1864	81(4)	220(9)	291(11)	166(9)	158(18)	53(5)	85(3)
T	54	T,t	48	7d,3d	RR0082-636	85(4)	240(20)	230(30)	68(18)	176(26)	58(11)	86(5)
t	13	t	11	5d	UR0020-a9	81(6)	195(20)	59(13)	59(15)	136(7)	48(5)	84(8)
u	43	u	39	1e	UR0035-2665	81(3)	210(18)	286(19)	249(17)	85(12)	167(9)	85(3)
u'	16	u	13		RR0082-877	84(10)	230(23)	298(13)	291(11)	159(11)	182(12)	84(9)
i	522	i	517	1c,1f	UR0020-a28	81(5)	203(19)	281(24)	149(18)	203(24)	177(13)	84(5)
i'	8	i	8		PR0018-n69	87(8)	64(14)	293(19)	152(16)	184(30)	166(13)	89(4)
L	14	L	11	5j	AR0027-b17	96(18)	218(13)	73(15)	63(13)	104(17)	178(6)	83(3)
n	273	n	246	1b,1k	AR0023-b71	84(5)	217(12)	289(10)	302(11)	187(22)	57(9)	142(10)
E	12	E	12	3b	RR0082-904	84(2)	217(15)	169(15)	280(19)	163(18)	49(9)	144(6)
E'	10	E	9		PR0057-t9	80(2)	227(14)	230(11)	290(14)	206(19)	49(10)	142(5)
g	19	g	9	1z	RR0082-1771	84(3)	207(9)	270(13)	197(18)	154(20)	53(8)	142(11)
s	43	s	42	5z	UR0026-3654	83(3)	207(6)	50(10)	164(7)	150(15)	51(5)	146(7)
m	11	m	9	7p	PR0033-b8	83(3)	229(20)	228(14)	67(9)	168(4)	55(6)	145(6)
11222	28	11222	23	1t	PTE003-b907	83(5)	208(17)	290(15)	181(19)	184(23)	183(15)	143(10)
d	9	d	8	7r	RR0082-262	81(3)	218(8)	217(17)	55(11)	172(14)	296(3)	151(5)
F	108	F	97	2a	RR0082-1711	142(10)	261(13)	291(20)	286(15)	192(17)	53(8)	84(4)
F'	12	F	6		RR0082-1879	144(8)	192(9)	244(11)	287(21)	171(12)	51(7)	7 9(5)
A	112	A	101	4a,0a,#a	RR0082-2485	148(9)	228(23)	127(36)	280(18)	156(18)	43(10)	87(6)
b	25	b/p	21	4g	UR0012-a226	147(9)	256(19)	172(25)	205(14)	165(15)	49(7)	83(3)
f	26	f	14	8d,4d	RR0009-c1062	148(5)	265(15)	220(32)	75(17)	186(25)	56(8)	87(4)
f'	30	f	26	6d	RR0082-1879	140(13)	235(26)	87(21)	58(14)	159(23)	50(8)	84(4)
l	19	l	14	4n	RR0082-767	143(10)	226(13)	203(20)	74(10)	215(20)	193(11)	83(4)
l'	56	l	43	0i,6n	RR0082-940	143(11)	266(21)	84(18)	78(25)	201(31)	181(11)	89(14)
r	58	r	44	2k	RR0082-264	143(11)	252(24)	292(23)	284(23)	209(20)	55(11)	145(8)
R	72	R	68	4b,0b	RR0082-247	144(8)	237(24)	147(31)	297(24)	168(18)	45(13)	144(7)
c	47	c	37	6p	RR0082-96	145(8)	260(19)	93(24)	73(18)	176(22)	54(14)	147(6)
h	9	h	9	4s	UR0026-2655	150(2)	248(14)	170(10)	277(12)	84(6)	177(5)	148(9)
12231	14	12231	2		RR0082-2613	106(29)	230(44)	71(20)	198(38)	256(29)	278(34)	104(23)
y	50	y	38		RR0082-1206	83(5)	221(13)	266(11)	60(17)	196(23)	279(22)	96(9)

as described in Figure 4.28-b. The exact parameters used for this coordinate system will be different according to the nature of the base doublet interaction. Thus, base pair geometries will be represented in a two-dimensional space, whereas base stacking geometries will be described using three parameters. Details will be given in Section 4.5.2 and 4.5.3. In the remainder of this chapter, we refer to this parametrization as the *center of pyrimidine* (COP) parametrization.

Both parameterizations present certain advantages and drawbacks. As we aim to build a method for the classification of base doublet interaction, it is important to know which coordinate system to use and in what case. For this reason, we now list some of the characteristics of the COM coordinate system and compare it to the proposed system.

As illustrated in Figure 4.28-a, the COM coordinate system employs the angle of rotation around the center of mass needed to align the two bases and the (x, y)-projection of the distance between the two glycosidic nitrogen on the plane of the lower base. Note that the glycosidic atom is the closest atom to the backbone. Hence, a classification method based on the glycosidic distance is expected to give better correlation with one based on backbone classification. Also, the relative location of two glycosidic atoms is fixed for any double helix geometry. Thus, this classification should be effective for detecting any possible deviations from the double helix structure.

A drawback of this method is that the translation and the rotation are not performed within the same coordinate system. The rotational coordinate system is based on the actual center of mass and the location of this origin depends on the type of base. The COP parametrization employs the same coordinate system for both rotation and translation, whose origin is set at the location of a pseudo-atom (pyrimidine ring geometrical center) rather a real atom. The pyrimidine ring seems to be involved in the majority of base pair and base stacking interactions. Hence, the COP coordinate system seems to be useful for base pair and base stacking classifications.

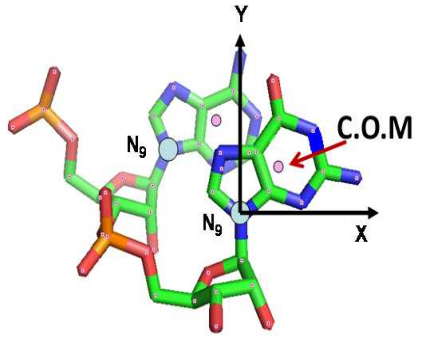
However, a disadvantage of this choice of coordinates is the possibility of certain artifacts in the classification between the purine and pyrimidine bases. Further, because the COP origin may be far from the backbone, we may have a weaker correlation with the backbone geometry.

4.5.2 Base Pair Geometry

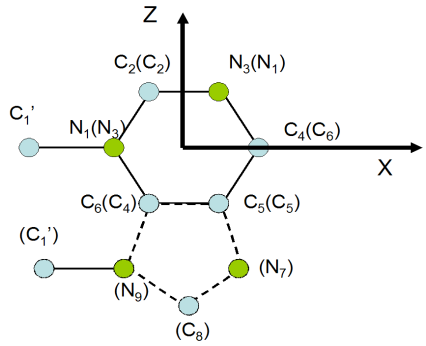
In this part, we focus on the analysis of base pair interactions. The most familiar and common case of such interactions is that of the Watson-Crick (W-C) base pairs, which is responsible for the double helical structure of polynucleic acids. Base pair geometry is flat, so that two bases lie approximately in the same plane. The base pair interaction is mediated by hydrogen bonds. Hydrogen bonds can form between an electronegative atom (electron "donor") and a hydrogen atom bonded to another electronegative atom (electron "acceptor"), both located at specific sites in the base [29]. The relative arrangement of the donor to the acceptor is such that each base possesses three possible edges for base pair interaction [62] (see Figure 4.29). This arrangement provides six different potential geometries between interacting bases. An additional combinatorial factor of two emerges from the directionality of the strands. This gives a total of twelve classes of base pair geometries, which are referenced in data banks using the Leontis-Westhof (LW) notation [62].

An automated classification method for the base pair geometry based on an Expectation Maximization (EM) algorithm was presented in [61]. In this method, Lemieux *et al.* proposed an elaborate classification of the base pair geometries. They show the existence of intermediate base pair geometries, but their work mainly agrees with the LW classification. The drawback of the EM method is that it required an assumption about the structure of the underlying distribution function (N Gaussians where N had to be pre-defined). This disqualifies the method for our purposes, since we are interested in a minimally-supervised classification method, where no prior

knowledge about the underlying distribution function is needed. In another work, Sarver *et al.* [84] have performed a detailed by-eye classification of all existing base pair geometries. This analysis was run using the COM parametrization.



(a)



(b)

Figure 4.28: (a) The center of mass parametrization for base pair interactions uses the rotation angle around the center of mass (red circle) that is needed to align the two bases and the (x, y)-projection of the distance between the two glycosidic nitrogen (blue circle) on the plane of the lower base. (b) COP coordinate system in the plane of the base.

We now describe in details the parameters that were used in the COP coordinate system in order to describe the base pair geometry.

As mentioned above, the COP parametrization used in this work differs according to the type of the base doublet interaction (base pair or base stacking). The relative geometry of the base pair can be represented by four parameters. Three of them form the directed (vector) distance between the centers of the two hexagonal pyrimidine rings using spherical coordinates, i.e., distance r and polar angles θ and ϕ (Figure

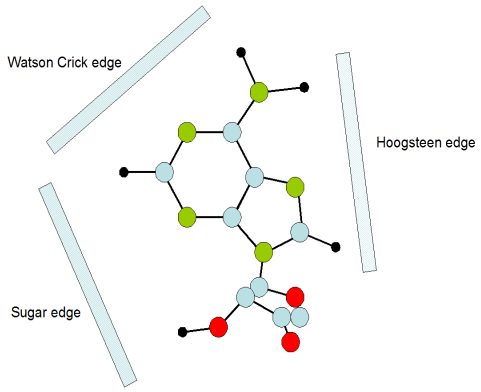


Figure 4.29: Possible edges for base pair interaction in a base.

1-b). Another angle ω accounts for the rotation required to align the two pyrimidine rings. We have reduced the number of parameters to two, by taking into consideration the fact that the two bases are coplanar and by eliminating the distance parameter. Hence, we use only θ and ω as the set of parameters for the clustering problem. We have run our code on the RR0082 (PDB number: 1S72) [53], i.e. the same as for the test case of the Sarver base pair classification [84].

The Potts method was applied to classify base pair geometries, using both COM and COP parameterizations.

For base pair geometry classifications, we found that both coordinate systems worked similarly, and so for simplicity we will just use the COP in what follows. The main point is to highlight the Potts method.

The list of base pairs was made by choosing base doublets that fulfill the following constraints: i) The distance between pyrimidine centers is less than $8.0\mathring{A}$; ii) the minimum distance between two atoms from the two bases is less than $3.5\mathring{A}$; iii) the angle ϕ is less then 115 degrees and larger then 65 degrees; and iv) the normals to the two bases form an angle less than 30 degrees.

In our representation the relative geometry of base pairs is defined by the bases only and not by the strand geometry. Each base, being flat, has two faces: up and down. We have chosen the relative direction of the two bases to be the preliminary

criterion for partition of the base pairs. Therefore we work with two groups of data: a group with the same directionality for both bases (up-up or down-down) and a group of opposite orientation of the faces (up-down or down-up). For the base of a nucleotide in a helix with the glycosidic torsion angle in *anti*, the normal to the up face is pointing in the $5'$ direction and the normal to the down face is pointing in the $3'$ direction.

We have classified these two groups into clusters using a Potts model clustering method. Since the problem is two dimensional (θ and ω), we chose the number of nearest neighbors parameter to be $K = 6$.

The susceptibility graph for the up-down group exhibits one dominant peak, which appears at a very low temperature ($T < 10^{-4}$). The second peak corresponds to the melting point for all the clusters (the second one ending the phase of interest). The up-up group shows one significant peak with a slow decaying tail. Again, as for the up-down case, the separation into clusters appears almost instantaneously ($T = 0$). Therefore, we have chosen the clustering temperature to be $T = 0.001$ for both up-down and up-up cases. This seems to give good results in comparison to the reference classification by Sarver *et al.* [84]. For the case of the up-up group, the existence of a second peak made us consider the clustering configuration at $T = 0.019$, for which an additional cluster is detected.

Figure 4.30 gives the two dimensional scatter plot for the up-down configurations and exhibits the major clusters formed by the Potts model based classification. Some validation can me made using a well-known measure of structural similarity within and between clusters. Thus, we computed the root mean square standard deviation (RMSD) for points inside each cluster and between points of two different clusters. For this base pair two-dimensional problem, we computed the following in order to

estimate the "intra-cluster" RMSD for each cluster C with n points:

$$\sqrt{\left(\sum_{i\in C}\sum_{j\in C}(\theta_i-\theta_j)^2+(\omega_i-\omega_j)^2\right)/(n(n-1)/2)} \tag{4.61}$$

. Then, for each pair of clusters (C_1, C_2), with n_1 and n_2 points respectively, we computed the "inter-cluster" RMSD:

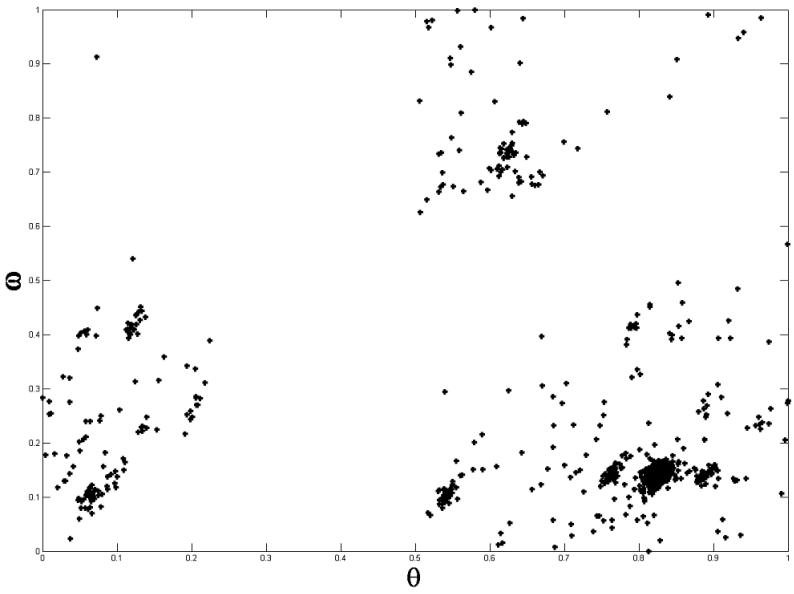
$$\sqrt{\left(\sum_{i\in C_1}\sum_{j\in C_2}(\theta_i-\theta_j)^2+(\omega_i-\omega_j)^2\right)/(n_1\cdot n_2)} \tag{4.62}$$

When comparing those quantities, we can see that, on average, the "intra-cluster" distances are ten times smaller than the "inter-cluster" ones.

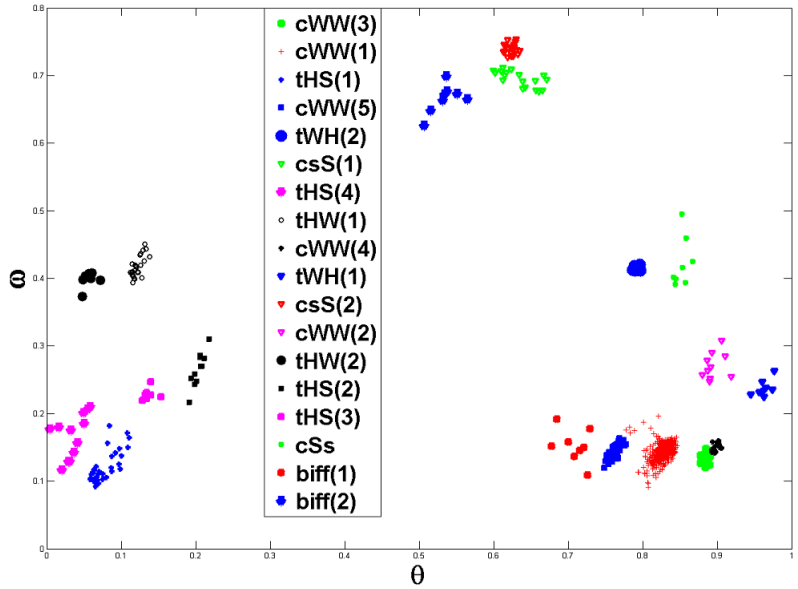
The clusters that emerged from the Potts classification are presented in Table 4.6 for the up-down case and in Table 4.7 for the up-up case.

The up-down class contains the majority of the data points (1171 cases from the total of 1406 base pair candidates). The first column gives the classification obtained with the Sarver *et al.* method [84] and the third one presents the results for the Potts classification. As observed in Table 4.6, the most predominant difference between our classification and the Sarver classification is that our classification splits most of the LW groups into sub-clusters.

Specifically, our overall framework can differentiate among *base pair contents*. By *base pair contents*, we refer to doublets $(Base_1 - Base_2)$, where $Base_1$ and $Base_2$ indicate the family of the two bases involved in the base pair. The base family is either purine or pyrimidine, so that one distinguishes four different cases of base pairs: $(pyrimidine - purine)$, $(purine - pyrimidine)$, $(pyrimidine - pyrimidine)$, $(purine - purine)$. Thus, the extra splitting that is observed for both the csS and $biff$ clusters [84] (Table 4.6) seems to illustrate this differentiation. However, we showed that, for most cases, extra splitting is not an artifact of the choice of coordinate system. If this were the case, the clusters would just correspond to distinct base pair contents. For example, for the tHS LW group, our method finds four distinct



(a)



(b)

Figure 4.30: 2D projection of the clustering for the base pair geometry case with up-down configuration. The coordinates of the projection are the normalized azimuthal and the rotational angles needed to align the two bases. (a) All the data points before clustering. (b) The major clusters obtained with the proposed method and that correspond to the LW classification ([63]). See Table IV for more details.

clusters. Some of these clusters exhibit the same base pair contents (e.g., AG) but are differentiated by the geometry.

Also, the subdivision of the cWW cluster to five sub-clusters seems to result from the differentiation between the standard Watson-Crick base pair geometry and the GU (or UG) base pair. All other divisions seem to be unrelated to the content groups. Since these groups are rather small (e.g., $cWW(2)$ in Table 4.6, one must study a much larger data set to verify (or rule out) this subdivision. Two of the LW groups, cWS and CSW, which do appear in the classification proposed by Sarver *et al.* [84], cannot be recognized as clusters by our method.

When ignoring the extra subdivisions that are obtained in the proposed classification, comparing the number of points in each one of the clusters that are also obtained by Sarver *et al.*, seems to give an excellent match. Examining the results for the up-up group as presented in Table 4.7 gives very similar conclusions. An important conclusion from this section is that our methodology is able to reproduce the LW classification [84], and also to give some additional information.

4.5.3 Base Stacking Geometry

The majority of bases in the RNA are arranged in stacks (see Figure 4.32). This stacking behavior is typical of aromatic rings [82]. Stacking interactions constitute a major factor in the stability of RNA helices and their assembly [63]. The exact nature of stacking forces between the RNA bases is still an open question. As far as we know, the possible sources of stacking interactions are: π stacking [100], dispersion, electrostatics [37, 92], dipole-induced-dipole interactions [12] and hydrophobic forces [31, 67]. The lack of a known physical model for base stacking geometries prevents us from using prior references, such as the one we had for base pair geometries or backbone classification. However, the major role that base stacking interaction plays in the folding and stabilization of the 3D RNA structure seems to be a strong incentive

Table 4.6: Up-down base pairs in RR0082 conformations

Cluster name	Cluster size	Sarver's group	Sarver's group size	Typical content
cWW(1)	671	cWW	668	
cWW(2)	10	cWW	0	GC basepairs (θ is larger than cluster 2)
cWW(3)	19	cWW	19	GU base-pairs
cWW(4)	19	cWW	19	GU base-pairs θ is larger than cluster 1
cWW(5)	42	cWW	42	36*UG base-pairs + 6*UU base-pairs(cwW)
tHW(1)	26	tHW	26	almost all base-pairs are AU
tHW(2)	7	tHW	6	almost all base-pairs are AC
tWH(1)	8	tWH	7	CA or AA base-pairs
tWH(2)	7	tWH	7	UA base-pairs
tHS(1)	35	tHS	34	AG base-pairs (AN6-GO2')
tHS(2)	11	tHS	11	AG base-pairs (AN6-GO4')
tHS(3)	7	tHS	0	AG N2-O2p interaction (i,i+3)
tHS(4)	12	tHS	8	pyr-pur,pur-pyr
tSH	43	tSH	42	AG base-pairs
csS(1)	19	CsS, CSs	12,5	pur-pur(CsS) or UA (cSs)
csS(2)	21	CsS, CSs	8,9	AC,CA,AU base-pairs
CSs	8	cSs	6	AG,GG
biff(1)	8	biff	8	CC base-pair
biff(2)	8	biff	2	AC,CA,UA base-pairs
cWS	–	-	13	
Csw	–	-	21	

Table 4.7: Up-up base pairs in RR0082 conformations

Cluster name	Cluster size	Sarver's group	Sarver's group size	Typical content
tWW	20	tWW	20	mostly pur-pyr
cWh	6	cWh	5	mostly GU basepairs
cHW	37	cHW,cHS	12,11	T=0.019 two groups that match the ZL notation. cHS mostly pur-pyr Mostly,(i,i+1+,(i,i+2),(i,i+3)
cHS(1)	11	cHS	11	UG base-pairs
cHS(2)	8	cHS	6	mostly pur-pur base-pairs
cSH	6	cSH	1	other base-pairs do match the tSH geometry
tWS	36	tWS	20	mostly AG base-pairs most of the non defined ZL cases also have tWS geometry
tSW	13	tSW	6	mostly AG base-pairs most of the non defined ZL cases also have tSW geometry
thH	18	thH,tHH	16,2	almost all AA base-pairs
tsS	17	tsS	16	almost all are GA base-pairs
tSs	37	tsS,tSs	14,20	mostly pur-pyr,AG base-pairs

to start developing such a geometry classification model.

Another benefit of finding a classification for base stacking geometries is the development of some structural annotation for bases in the same manner as for the backbones [41, 42]. To the best of our knowledge, the only attempt to perform a detailed and unsupervised classification of base stacking interaction was performed

by Sykes and Levitt [94]. In their work, a clustering scheme has been developed that consists of a mixture of k-means and agglomerative classifications. This hybrid algorithm was used to classify all types of base doublet interactions in RNA structures. The main drawback of the algorithm is that no criterion is proposed for choosing the level in the data decomposition at which one observes the optimal classification. Different optional clustering configurations are presented for different levels. Whenever it was possible, we compared our results for the base stacking with this method.

The Potts model was applied, for the base stacking case, using two alternative data sets, corresponding to two different parameterizations. These two parameterizations have been introduced in Section 4.5.2: that is the COM and the COP coordinate system. These two parameterizations give different *realizations* of the base stacking, and hence, can be used for cross-validation of the stacking scheme.

The data file that we used for our analysis was the same RR0033 structure that employed previously in this work. With the COM parametrization, we used a data set of base stacking cases that was provided by Leontis and Zirbel (private communication). For the COP coordinate system, the criteria used to select the base stacking doublets are the following: i) the two faces are "nearly" parallel in the sense that the angle between their respective normals is less than 30 degrees; ii) the vertical distance between two bases is 3-5$\mathring{A}$; iii) the distance between the centers of the pyrimidine rings is less then 8$\mathring{A}$; and iv) there are at least two atoms from the two bases within 3.5$\mathring{A}$ from one another.

It is important to note that the criteria of the two parametrization methods (i.e., COP and COM as explicated in [84]) are not identical. As a result, the two lists of base stacking doublets used for the Potts clustering were not identical. In fact, there was about a 10% difference in the content of both lists. We have applied the clustering method to both data sets rather than trying to find an identical criteria for validation purposes.

Note that a common problem of various clustering techniques is the lack of robustness [47]. Thus, a small change in a data set can cause a large change in the classification. Comparing the results for the two types of parametrizations may therefore give a measure for the robustness of the Potts method. The common clusters of both parametrization systems provides a strong argument to their validity.

The first coordinate system that we have utilized involves the COP parametrization, similarly to what has been done for the base pair analysis given above. However, contrary to base pair geometry, base stacking does not involve co-planar bases. Therefore, three parameters are needed (instead of two) to describe the base stacking geometry for the COP system. These three parameters are the projections of the distance vector between the centers of the pyrimidine rings onto the x-z plane (Figure 4.28-b) and the rotation angle ω (Figure 4.24-b). For the COM parametrization, the coordinate system is the one presented in the Section 4.5.2 for base pairs.

In the same manner as for the base pair set-up, there is a preliminary division among base stacking geometries based on the relative orientation of the bases. There are four possible arrangements for the faces of the base doublets: up-up, down-down, up-down and down-up. We therefore identify four different data sets.

The up-up and the down-down stacking geometries have the same type of interaction. The interacting faces are the upward face of the first nucleotide with the downward face of the second nucleotide in the up-up group, and vice versa for the down-down case. The difference between the two cases is that the up-up group is represented inside a double helix while the down-down group is represented outside of a Watson Crick helix. Hence the down-down base stacking geometries are free of packaging constraints as well as stacking cooperative effects, and show a different distribution (i.e., more dispersed distribution) in the configuration space. This reason led us to separate between the two groups.

Upon initial visual inspection, none of the four data spaces actually shows any

obvious clustering structure. For example, examination of the 3D data space of the up-up case (Figure 4.31) shows a very vague cluster structure probably due to the less restrictive nature of stacking forces compared to forces that determine base pair and backbone orientation. Inhomogeneity is not obvious here and clustering becomes more challenging.

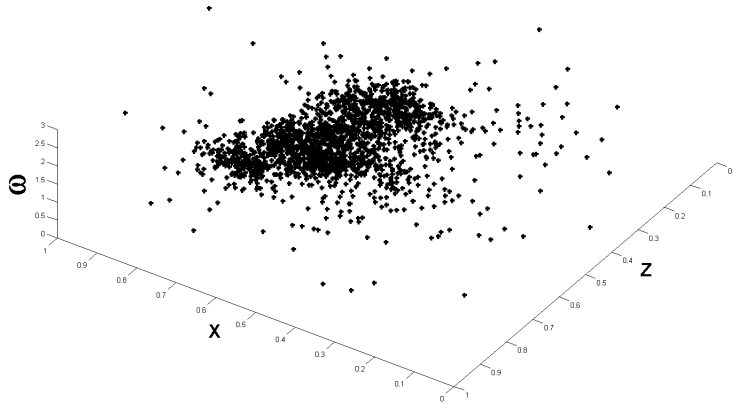


Figure 4.31: Data points for the base stacking, with up-up configuration and using the COP parametrization. Possible clusters are difficult to detect.

We have performed cluster analysis using the Potts algorithm on all the four cases: up-up, down-down, up-down and down-up. Given the two parameterizations COM and COP, we needed to run the Potts algorithm on eight different cases. For our analysis, we chose the scale parameter a to be equal to the average distance between neighbors and, according to the criterion defined in Section 4.3.1, $K = 12$. Also, since the dimensions along which the stacking geometries are analyzed involve both Euclidean distances and angles, we normalized every variable has been normalized between 0 and 1 before calculating distances d_{ij} and applying clustering. The criterion for choosing the clustering temperature stays the same as in Section 4.5.2. For six of the eight runs, a single temperature was found adequate to span all the clusters. Increasing the temperature beyond this reference either shrinks or melts clusters, while decreasing the temperature prompts merging of clusters. The only case where more than one temperature was required was for the up-up and up-down face groups

with the COP parametrization. These cases will be explained separately in the text.

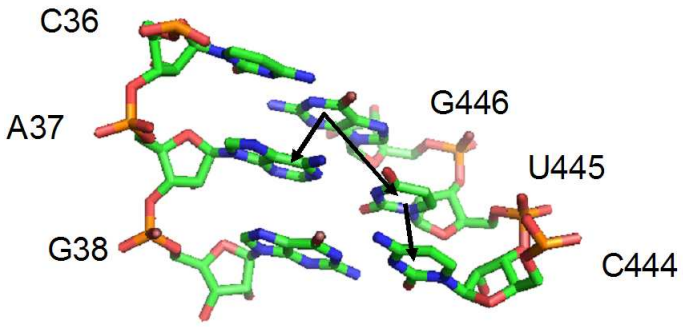


Figure 4.32: Double helix structure exhibiting base stacking structure. We can observe the different classes: I) Pyrimidine-purine, II) Pyrimidine-pyrimidine or Purine-purine and III) Purine-purine.

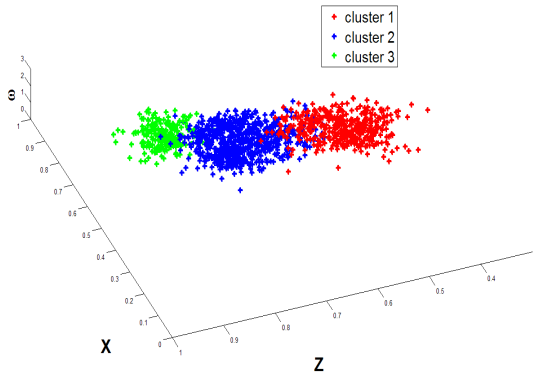


Figure 4.33: Three major clusters for the up-up case of base stacking in the COP parametrization.

4.5.3.1 Validation of Results

Given the lack of prior knowledge about the clusters appearing in the stacking geometry, we used the Watson-Crick double helical structure as the basis for validation of our analysis. Thus, we first validate our clustering results on the only data points that exhibit this structure. We define a Watson-Crick double helical structure to be a double strand consisting of a contingent of two or more base pairs with a Watson-Crick geometry [82].

The double helical arrangement is the most abundant motif in the RNA structure. About half of base stacking interactions that are candidates for classification lie within the Watson-Crick double helix definition. A typical Watson-Crick helix is shown in (Figure 4.32), demonstrating that the geometry of the intra-strand stacking doublets depends upon stacked nucleotide pairs.

We distinguish among three different classes: pyrimidine-purine (Class I), purine-purine or pyrimidine-pyrimidine (Class II) and purine-pyrimidine (Class III). We will refer to these as "content classes." It is important to note that there is no sequence symmetry in RNA, and that the numerical order (5′-to-3′ and 3′-to-5′) of bases is important. This ordering stems from the fact that each base has two different faces. The most common intra-strand face arrangement in the Watson-Crick double helix is up-up, hence, we have chosen this class to be the major validation case for the clustering application.

The validation of the content classes can be done only with the COP parametrization, since the COM technique was constructed to group all stacking cases of a Watson-Crick double helix into one class. Indeed, performing clustering with the COM parametrization only reveals one cluster for all the A-form RNA double helix stacking conformations. On the other hand, the COP parametrization reveals three clusters that clearly characterize the three content classes.

For the up-up group, Table 4.8 gives the comparison between the three content classes and the three major clusters obtained with the Potts algorithm at $T = 0.13$. We use the term "majority cluster" to name the cluster that overlaps the most with the considered content class. As can be seen from Table 4.8, about 90% of the class I group that is in a Watson-Crick helix is also in cluster 1. The same observation is true for class II and cluster 2. As for class III, only 70% are in cluster 3. We have performed a similar cluster analysis when removing the Watson-Crick double helix constraint (i.e. when accounting for all data points, even the non-Watson-Crick

Table 4.8: Up-up group in Watson-Crick double helix content groups

Content class	majority cluster	# of bases in content class	# of bases in majority cluster
pyr-pyr	1	597	570
pur-pyr	2	210	178
pyr-pur	3	347	310

Table 4.9: Up-down group in Watson Crick double helix content groups

Content class	majority cluster	# of bases in content class	# of bases in majority cluster
pur-pur	1	130	111
pyr-pyr	2	73	67
pyr-pur	3	309	289

doublets) and have observed similar results.

Agreement of our clustering results with base content is very good. Exceptional cases, i.e. those for which the two classifications diverge, might represent the most interesting but non-trivial cases of base stacking. These will probably include stacking of uniaxial helices, junctions, and bulges [54]. These non-trivial motifs define the final 3D structure of RNA. Correct classification of this kind of stacking would help in the understanding of the folding and the self-assembly of helical motifs into a functional well-defined 3D structure.

Examination of the data space in the up-up group demonstrates the power of the Potts clustering method. Figure 4.31 shows a 3D projection of that data space without any filtering and reveals no clear underlying cluster structure at first sight. Using the Potts classifier as a filter to select the most populated clusters gives very satisfying results (see Figure 4.33).

The next largest data group is the up-down group. Members of this group participate in the inter-strand Watson-Crick double helical structure, as shown in Figure 4.32. Following the usual procedure, we produced and analyzed our clustering

results at temperature $T = 0.12$. We observe three major clusters that can be easily affiliated with three "content classes." Quantitative results for these three clusters are shown in Tables 4.9. The two tables show the same types of results as for the up-up case.

4.5.3.2 New Clusters

After validating the reliability of the algorithm (for the case of the COP parametrization), we have performed a full scan that was intended to find new non-trivial clusters in the RR0033 data base. The clustering was performed with both parametrization methods. The results of this clustering procedure are presented in Table VIII.

The table describes the different clusters according to their affiliation to the face-face group. In this table, we provide the total number of base stacking doublets in the corresponding face-face group for both the COP and COM parameterizations. These numbers are a bit different because the criteria for base stacking are slightly different for the two representations. For each face-face group, the temperature that was used for the classification in both representations is also given. The next part of the table includes the specific clusters that have been identified. The first column of this part includes the cluster numerical annotation. We report the number of points (stacking doublets) that were identified in each cluster, as well as the number of points that are common to a given cluster in both representations.

It should be noted that only for the first cluster in the up-up group is the number of doublets in the COM parametrization much larger than the number of doublets in the COP representation. This discrepancy is due to the fact that the first cluster represents all of the intra-strand stacking conformations in a Watson Crick helix. As mentioned previously in section 4.5.2, the COM parametrization was designed to provide a good way to define the constraints of a Watson-Crick double helical arrangement of bases in the RNA.

Table 4.10: Clusters of base stacking doublets

Base orientation group	Points w/ C.o.P.	Points w/ C.o.M.	Temp w/ C.o.P.	Temp w/ C.o.M.	Cluster index	Points in C.o.P. cluster	Points in C.o.M. cluster	Common Points	Typical LW content	Base type	Geometry	Sequence order
up-up	1902	2128	0.001	0.006	1	1818	1991	1740	(cWW,cWW)			
					2		27		9 × (cWW,tWH)	pur-pur	im(i) above im(j)	
down-down	94	87	0.12	0.129	3	23	17	14	10 × (cWW,cWW)		Im(i) above sugar(j)	
					4	32	30	26	(tWS,*), (tWH,*)		pyr(i) above pyr(j)	
up-down	858	715	0.111	0.098	5	-	19	-	8 × (tWH,*)	10*UA	Im(i) above O2(j)	16 × (i+2,i)
					6	19	10	5	7 × (tHS,cWW)	pur-pur	pyr(i) above N6/O6(j)	
					7	527	465	398	(cWW,cWW)	pur-pur		
down-up	215	178	0.089	0.105	8	19	31	17	10 × (tHH,tHS/tWS)	AC or AU	pyr(i) above N6/O6(j)	28 × (i+1,i)
					8a	10	same cluster	9		pyr-pyr		
					9	22	38	16	7 × (tHS,*)	AA, pyr-pur, pur-pyr		
					9a	13	same cluster	10	7 × (tHS,tWW)	GG,AG,UG	pyr(i) above pyr(j)	
					10	55	-	-	(cWW,cWW)	pur-pur		far appart

The last three columns of the table give details on the characteristic features of each stack class. The first of these columns gives the base pair arrangement that each one of the residues is involved in (if such an interaction exists). The base pair interactions are arranged in the $3'$-$5'$ order and the annotation that we are using is the same as in Section 4.5.2. The next column gives the majority content class of the doublets contained in the corresponding cluster. In the last column, we describe a typical geometrical feature of the cluster. In this table, we also have included clusters that are parametrization specific. Cluster 10, the last cluster of the down-up group, has base doublets that do not qualify to be base pairs by the COM classification, because the centers of mass of the two bases are two far apart to be defined as a base stack doublet by this technique. Cluster 2, found with the COM parametrization, contains many doublets that do not appear in the COP parametrization because of the sparsity of data points in this cluster. The same reasoning can be used to explain the failure of the Potts classification to find cluster 5 using the COP coordinates. Clusters 8 and 9 using COM are split into two clusters in the COP parametrization. Some differences in the content of the split clusters (8 vs $8a$ and 9 vs $9a$) seems to indicate that the extra clustering is necessary. At this stage, we have chosen to not consider the splitting, due to the small number of members in the split clusters. Finally, we would like to note that some of the new clusters are characterized by non-traditional stacking arrangements. While we have not confirmed in this manuscript that all these clusters represent a preferred energy state, we nevertheless believe that these types of stacking are not artifacts and do not represent arbitrary arrangements. This subject will be the focus of future research. Figure 4.34 shows a typical stacking doublet from each one of the non-trivial (non-Watson-Crick) clusters. We have identified nine new clusters, among which seven appear in both parameterizations.

4.5.3.3 Comparison to Other Clustering Schemes and Past Classifications

These results have been compared to the classification that was suggested by Sykes' work [94], which, as mentioned earlier, proposed to use a mixture of k-means and agglomerative clustering methods. We have found that all of the representative doublets from the different stacking clusters discovered in Sykes' work belonged to the up-up and up-down trivial (Watson-Crick) stacking. This finding is not surprising, since methods based on k-means are known to perform over-classification of the densely populated regions of the data space [41].

Furthermore, the results for base stacking geometries have been compared to those obtained with classical linkage methods. As mentioned in Section 4.3.2, a criterion needs to be found in order to know at which level of the hierarchical decomposition one should perform clustering analysis. An informal indicator of the "best number of clusters has been proposed by Calinksi *et al.* [14]. This criterion is referred to as the *variance ratio criterion* (VRC). For a given number of clusters M, the N data points are distributed among clusters by applying the criterion of minimum within-cluster sum of squares (WGSS) and maximum between-clusters sum of squares (BGSS). Once these optimal quantities have been found, the corresponding VRC can be calculated as follows:

$$VRC(M) = \frac{BGSS}{M-1} / \frac{WGSS}{N-1} \tag{4.63}$$

This ratio is analogous to the F-statistic in univariate analysis. However, the theoretical significance of this ratio is not intuitive. The important point here consists in considering this quantity as an estimation of the goodness of the "split". This can thus be seen as a way to compare the optimal clustering configurations that have been obtained with different number of clusters M. Therefore, by choosing the number M for which the VCR reaches a local, if not global, maximum. In this work, this

criterion has been used in order to determine at which level of a dendrogram clustering configurations should be analyzed when using linkage methods. Of course, it is important to mention that this constitutes an extension, and probably an estimation of the optimal number of clusters, since each level of a given dendrogram does not necessarily satisfy the "minimum within-cluster sum of squares" criterion. Yet, using this VRC criterion rule helped perform a classification of base-stacking geometries using linkage clustering. Thus, for each data set, the VRC ratio has been computed for all levels of the corresponding dendrogram and a VRC graph has been established in order to determine the "best" number of clusters. This graph sometimes exhibits several "peaks" that can be seen as local maxima. In these cases, the choice of the number of clusters turns out to be complicated. However, results often show a pretty good correspondence with the Potts model classifications. Future research on the use of this VRC criterion for hierarchical clustering may certainly yield interesting outcomes.

4.6 Concluding Remarks

The work presented in this chapter proposed a clustering algorithm based on a Potts model. This algorithm was used to validate two documented structures, namely the single nucleotide conformation and the base pair geometries, and one undocumented structure, namely the base stacking. For the documented cases we obtained reasonably good classifications by using the clustering procedure in a fully automated manner without employing any prior knowledge. However, some discrepancies between our clustering results and the previously-published classifications were observed. Some of these turn out to appear in the fringes of the classes and do not seem to pose real issues. In several other cases, the discrepancies lead to the merging or splitting of clusters. By comparing these results to well-known structural motifs, we can conclude that those obtained with the Potts model seem to be finer than the

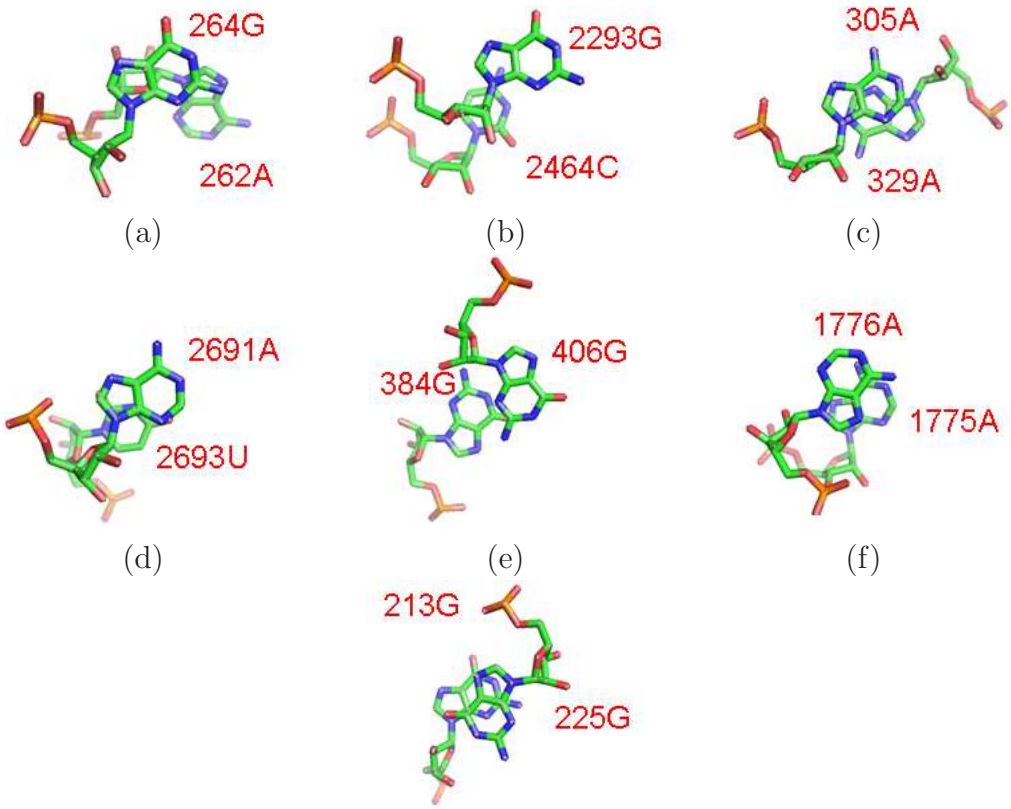


Figure 4.34: In this figure, we show examples of the seven new stacking geometries that were discovered using the Potts classification. Figure (a) gives the single new up-up cluster (**cluster** 2). Figure (b) gives the first new cluster of the down-down group. Figure (c) gives the second cluster of the same group (**cluster** 4). Typical geometries from the first (**cluster** 5) and the second (**cluster** 6) clusters of the up-down group are respectively shown in figure (d) and figure (e). Typical geometries from the first (**cluster** 8) and the second (**cluster** 9) clusters of the down-up group are respectively shown in figure (f) and figure (g).

previously-developed classifications. For the case of base stacking we have established and validated a new classification scheme. This classification can be used as a base for a new structural annotation of RNA structures, which will enable a complete description of the backbone and base pair annotation schemes.

Finally, we have demonstrated the ease of use of our proposed method. We have shown through our examples that only one or two "temperatures" are typically needed for the analysis. The only cases that may require prior knowledge remain those for

which the susceptibility diagram exhibits more than one peak. There, the more challenging step consists in determining what temperature defines the best clustering configuration. Despite this degree of uncertainty, we have demonstrated in the present work that this process constitutes a much simpler task than the choice of an optimal cut in a hierarchical clustering tree.

CHAPTER V

CONCLUDING REMARKS AND FUTURE RESEARCH

In this thesis, two very different examples of statistical feature extraction and data analysis have been presented.

In the first part of the thesis, a wavelet-based model for representation, compression and de-noising of 3D surfaces was proposed. First, it was shown that second generation wavelets could be used to encode genus-one surfaces that are represented by triangulated meshes. Similarly to the methodology developed by Nain *et al.* in the context of spherical shape analysis, the proposed framework is decomposed as follows: a surface is first equipped with a regular multi-scale grid, wavelet basis functions are then built on this regular mesh structure, and the spatial coordinates of the surface vertices are finally encoded by projection onto this set of basis functions. Second, aligned with the idea of developing an efficient and robust surface encoding methodology, a wavelet-based algorithm was proposed to address the problem of shape compression and de-noising. The wavelet shrinkage framework that was developed in this thesis allowed us to efficiently remove noise-like wavelet coefficients and, thus, to appropriately smooth and compress the original shape signal. This data-driven statistical model is capable of locally controlling the strength of the *shrinkage* by accounting for spatial and inter-scale correlation between wavelet coefficients.

In this work, triangulated surfaces have been analyzed. A surface constitutes the boundary of a 3D object and its smoothness usually remains quite sensitive to noise. Thus, possible directions for continuing this research on multi-scale shape analysis would include volume-based shape representation. By defining a way to encode the whole volume of an object using multi-scale tools, similar models for enhancement, de-noising and compression could potentially be developed and applied to 3D models.

In the second part of this thesis, a non-parametric clustering method, based on a Potts model, was applied to RNA conformation classification. The clustering algorithm was used to validate two documented structures, namely the single nucleotide conformation and the base pair geometries, and one undocumented structure, namely the base stacking. First, for the documented cases, reasonably good classifications were obtained when using the clustering procedure in a fully automated manner (with no need for any prior knowledge). By comparing these results to well-known structural motifs, we were able to conclude that the Potts model leads to finer classifications than the previously-developed methodologies. Next, for the case of base stacking, we have established and validated a new classification scheme. This classification, along with the description of the backbone and base pair annotation schemes, may be used as a base for a new structural annotation of RNA structures. Possible directions for future research would include conducting further analysis of the "new clusters" that were found in the proposed classification. The validation of the content of these clusters would help confirm that the corresponding conformations actually are preferred spatial arrangements in the RNA structure. Finally, we have demonstrated the ease of use of our proposed method. Susceptibility diagrams provide an efficient tool for following the hierarchical decomposition of the data set into clusters. In most cases, the "temperature" that corresponds to the optimal arrangement of clusters is easy to detect. The only cases that may require prior knowledge remain those for which the susceptibility diagram exhibits more than one peak. There, the more challenging step consists of determining what temperature defines the best clustering configuration. Despite this degree of uncertainty, we have shown in the present work that this process constitutes a much simpler task than the choice of an optimal cut in a hierarchical clustering tree.

In future research projects, an extension to this clustering model might include the development of a methodology for analyzing the shape of the clusters in the

data space. Indeed, our clustering results show that the spatial configuration of a cluster may vary, from a multivariate Gaussian distribution to a very complex multi-mode shape. It would therefore be interesting to understand, analyze and encode the information carried by these shape characteristics. Thus, multi-scale shape analysis may find an unexplored application in this direction. Once a cluster is found and its boundary precisely defined, shape analysis tools, such as spherical harmonics or spherical wavelets, may be used in order to model and analyze the spatial structure of the cluster.

REFERENCES

[1] "Stanford bunny, the stanford 3d scanning repository."

[2] Abramovic, F. and Benjamini, Y., "Adaptive thresholding of wavelet coefficients," *Computational Statistics and Data Analysis*, vol. 22, pp. 351–361, 1996.

[3] Abramovic, F., Sapatinas, T., and Silverman, B., "Wavelet thresholding via a bayesian approach," *Journal of the Royal Statistical Society, Ser. B*, vol. 60, pp. 725–749, 1998.

[4] Agrawal, H. and Domany, E., "Potts ferromagnets on coexpressed gene networks: Identifying maximally stable partitions," *Physical Review Letters*, vol. 90, pp. 158102–158106, 2003.

[5] Angenent, S., Haker, S., Tannenbaum, A., and Kikinis, R., "Laplace-beltrami operator and brain flattening," *IEEE Trans. Med. Imag.*, vol. 18, pp. 700–711, 1999.

[6] Angenent, S., Haker, S., Tannenbaum, A., and Kikinis, R., "On area preserving mappings of minimal distortion," in *Internal Communication*, pp. 127–137, 2003.

[7] Ban, N., Nissen, P., Hansen, J., Moore, P., and Steitz, T., "The complete atomic structure of the large ribosomal subunit at 2.4 *aa* resolution," *Science*, vol. 289, pp. 905–919, 2000.

[8] Berman, H. M., Olson, W. K., Beveridge, D. L., Westbrook, J., Gelbin, A., Demeny, T., Hsieh, S.-H., Srinivasan, A. R., and Schneider, B., "The nucleic acid database: A comprehensive relational database of three-dimensional structures of nucleic acids," *Biophys. Journal*.

[9] Berman, H., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T., Weissig, H., Shindyalov, I., and Bourne, P., "The protein data bank," *Nucleic Acids Research*.

[10] Blatt, M., Wiseman, S., and Domany, E., "Data clustering using a model granular magnet," *Neural Computation*, vol. 9, pp. 1805–1842, 1997.

[11] Brenier, Y., "Polar factorization and monotone rearrangement of vector-valued functions," *Com. Pure Appl. Math.*, vol. 64, pp. 375–417, 1991.

[12] Bugg, C., Thomas, J., and Sundaralingam, M., "Stereochemistry of nucleic acids and their constituents. x. solid-slate base-slacking patterns in nucleic acid constituents and polynucleotides," *Biopolymers*, vol. 10, pp. 175–219, 1971.

[13] Cai, T. and Silverman, B., "Incorporating information on neighbouring coefficients into wavelet estimation," *Indian J. of Statist.*, vol. 63 (B), pp. 127–148, 2001.

[14] Calinski, T. and Harabasz, J., "A dendrite method for cluster analysis," *Comm. in Statist., Simul. and Comput.*, vol. 3, pp. 1–27, 1974.

[15] Cech, T., "Ribozyme, the first 20 years," *Biochem. Soc. Tran.*, vol. 30, pp. 1162–1166, 2001.

[16] Chang, S., Yu, B., and Vetterli, M., "Spatially adaptive wavelet thresholding with context modeling for image denoising," 1998.

[17] Chang, S., Yu, B., and Vetterli, M., "Adaptive wavelet thresholding for image denoising and compression," *IEEE Trans. Image Proc.*, vol. 9 (9), pp. 1532–1546, 2000.

[18] Chang, S., Yu, B., and Vetterli, M., "Spatially adaptive wavelet thresholding with context modeling for image denoising," *IEEE Trans. Image Proc.*, vol. 9 (9), pp. 1522–1531, 2000.

[19] Cremer, D. and Pople, J., "A general definition of ring puckering coordinates," *Journal of the American Chemical Society*, vol. 97, pp. 1354–1358, 1975.

[20] Delouille, V., Jansen, M., and von Sachs, R., "Second-generation wavelet denoising methods for irregularly spaced data in two dimensions," *Signal Processing*, vol. 86(7), pp. 1435–1450, 2006.

[21] Dempster, A., Laird, N., and Rubin, D., "Maximum likelihood from incomplete data via the em algorithm," *J. Royal Statist. Soc.*, vol. 39(1), pp. 1–38, 1977.

[22] Desbrun, M. and Cani-Gascuel, M., "Active implicit surface for computer animation," in *Proc. of Graph. Interf.*, pp. 143–150, 1998.

[23] Desbrun, M., Meyer, M., Schröder, P., and Barr, A., "Implicit fairing of irregular meshes using diffusion and curvature flow," in *SIGGRAPH '99 Conference Proc.*, pp. 317–324, 1999.

[24] Donoho, D. and Johnstone, I., "Ideal spatial adaptation by wavelet shrinkage," *Biometrika*, vol. 8, pp. 425–455, 1994.

[25] Donoho, D. and Johnstone, I., "Adapting to unknown smoothness via wavelet shrinkage," *J. Amer. Stat. Assoc.*, vol. 90, no. 432, pp. 1200–1224, 1995.

[26] Duarte, C. and Pyle, A., "Stepping through an RNA structure: A novel approach to conformational analysis," *Journal of Molecular Biology*, vol. 284, pp. 1465–1478, 1998.

[27] Duarte, C., Wadley, L., and Pyle, A., "RNA structure comparison, motif search and discovery using a reduced representation of RNA conformational space," *Nucleic Acids Research*, vol. 31, pp. 4755–4761, 2003.

[28] Eck, M., DeRose, T., Duchamp, T., Hoppe, H., Lounsbery, M., and Stuetzle, W., "Multiresolution analysis of arbitrary meshes," in *SIGGRAPH '95 Conference Proc.*, pp. 173–182, 1995.

[29] Feffrey, A., *An Introduction to Hydrogen Bonding*. Oxford University Press, 1997.

[30] Florian, P. and Domanyiegler, G., "Kissing numbers, sphere packing and some unexpected proofs," *Notices of the AMS*, vol. Sep, pp. 873–883, 2004.

[31] Friedman, R. and Honig, B., "A free energy anlaysis of nucleic acid base stacking in aquaeous solution," *Biophys. Journal*, vol. 69, pp. 1528–1535, 1995.

[32] Fukunaga, K., *Introduction to Statistical Pattern Recognition, 2nd edition*. Academic Press, New York, 1990.

[33] Gangbo, W. and McCann, R., "The geometry of optimal transportation," *Acta Math.*, vol. 177, pp. 113–161, 1996.

[34] Gautheret, D., Major, F., and Cedergren, R., "Modeling the three-dimensional structure of RNA using discrete nucleotide conformational sets," *Journal of Molecular Biology*, vol. 229, pp. 1049–1064, 1993.

[35] Gu, X. and Yau, S.-T., "Computing conformal structures of surfaces," *Comm. in Information and Systems*, vol. 2(2), pp. 121–146, 2002.

[36] Gu, X. and Yau, S.-T., "Global conformal surface parameterization," in *Eurographics Symps. on Geom. Process.*, pp. 127–137, 2003.

[37] Gupta, G. and Sasisekharan, V., "Theoretical calculations of base-base interactions in nucleic acids: Satcking interactions in free bases," *Nucleic Acid Research*, vol. 5, pp. 1639–1653, 1978.

[38] Haemer, M. D. and Zyda, M., "Simplification of objects rendered by polygonal approximations," *Comp. and Graph.*, vol. 15 (2), pp. 175–184, 1991.

[39] Haker, S., Angenent, S., Tannenbaum, A., and Kikinis, R., "Conformal surface parametrization for texture mapping," *IEEE Trans. Visual. and Comp. Graph.*, vol. 6(2), pp. 181–189, 2000.

[40] Haker, S., Angenent, S., Tannenbaum, A., and Kikinis, R., "Nondistorting flattening maps and the 3d visualization of colon ct images," *IEEE Trans. Med. Imag.*, vol. 19, pp. 665–670, 2000.

[41] Hershkovtis, E., Sapiro, G., Tannenbaum, A., and Williams, L., "Statistical analysis of RNA backbone," *IEEE Trans. on Computational Biology and Bioinformatics*, vol. 3, pp. 33–46, 2006.

[42] Hershkovtis, E., Tannenbaum, E., Howerton, S., Sheth, A., Tannenbaum, A., and Williams, L., "Automated identification of RNA conformational motifs: Theory and application to the hm lsu 23s rRNA," *Nucleic Acids Research*, vol. 1, pp. 6249–6257, 2003.

[43] Hilton, M. and Ogden, R., "Data analytic wavelet threshold selection in 2-d denoising," *IEEE Trans. Signal Proc.*, vol. 45, pp. 496–500, 1997.

[44] Hinker, P. and Hansen, C., "Geometric optimization," in *IEEE Visual. '93 Proc.*, pp. 189–195.

[45] Hoppe, H., DeRose, T., Duchamp, T., McDonald, J., and Stuetzle, W., "Mesh optimization," in *SIGGRAPH '93 Conference Proc.*, pp. 19–26, 1993.

[46] Horman, K. and Greiner, G., "Mips: An efficient global parameterization method," in *Curve and Surf. Design, St Malo*, pp. 153–162, 1999.

[47] Jain, A. and Dubes, R., *Algorithms for Clustering Data.* Prentice Hall Englewood Cliffs, New Jersey, 1988.

[48] Jansen, M. and Bultheel, A., "Empirical bayes approach to improve wavelet thresholding for image noise reduction," *J. of the Amer. Statist. Assoc.*, vol. 96(454), pp. 629–6397, 2001.

[49] Jansen, M., Malfait, M., and Bultheel, A., "Generalized cross validation for wavelet thresholding," *IEEE Trans. Signal Proc.*, vol. 56, pp. 33–44, 1997.

[50] Kalvin, A., Cutting, C., Haddad, B., and Noz, M., "Constructing topological connected surfaces for the comprehensive analysis of 3d medical structures," in *SPIE vol. 1445 Image Processing*, pp. 247–259.

[51] Kanai, T., Suzuki, H., and Kimura, F., "Metamorphosis of arbitrary triangular meshes," *IEEE Comp. Graph. and Appl.*, vol. 20(2), pp. 62–75, 2000.

[52] Kent, J., Parent, R., and Carlson, W., "Establishing correspondences by topological merging: a new approach to 3-d shape transformation," pp. 271–278, 1991.

[53] Klein, D., Moore, P., and Steitz, T., "The roles of ribosomal proteins in the structure assembly, and evolution of the large ribosomal subunit," *J. of Math. Biol.*, vol. 340, pp. 141–177, 2004.

[54] Klein, D., Schmeing, T., Moore, P., and Steitz, T., "The kink-turn: a new RNA secondary structure motif," *EMBO Journal*, vol. 20, pp. 4214–4221, 2001.

[55] Knott, M. and Smith, C., "On the optimal mapping of distributions," *J. Optim. Theory*, vol. 43, pp. 39–49, 1984.

[56] Lazarus, F. and Verroust, A., "Metamorphosis of cylinder-like objects," *J. Visual. and Comp. Anim.*, vol. 8(3), pp. 131–146, 1997.

[57] Lazarus, F. and Verroust, A., "Three-dimensional metamorphosis: a survey," *The Visual Computer*, vol. 14, pp. 373–389, 1998.

[58] Lee, A., Dobkin, D., Sweldens, W., and Schröder, P., "Multiresolution mesh morphing," in *SIGGRAPH '99 Conference Proc.*, pp. 343–350, 1999.

[59] LeFaucheur, X., Vidakovic, B., Nain, D., and Tannenbaum, A., "Bayesian spherical wavelet shrinkage: Applications to shape analysis," in *Proc. of SPIE Optics East*, 2007.

[60] Lemieux, F. and Major, F., "Automated extraction and classification of RNA tertiary structure cyclic motifs," *Nucleic Acids Research*, vol. 34, pp. 2340–2346, 2006.

[61] Lemieux, S. and Major, F., "RNA canonical and non-canonical base pairing types: a recognition method and complete repertoire," *Nucleic Acids Research*, vol. 30, pp. 4250–4263, 2002.

[62] Leontis, N. and E.Westof, "Geometric nomenclature and classification of RNA base pairs," RNA, vol. 7, pp. 499–512, 2001.

[63] Leontis, N. and Westof, E., "Analysis of RNA motifs," *Curr. Op. Struct. Biol.*, vol. 13, pp. 300–308, 2003.

[64] Levy, B., "Constrained texture mapping for polygonal meshes," pp. 417–424, 2001.

[65] Levy, B. and Mallet, J., "Non-distorted texture mapping for sheared triangulated meshes," pp. 343–352, 1998.

[66] Lounsbery, M., DeRose, T., and Warren, J., "Multiresolution analysis for surfaces of arbitrary topological type," *ACM Trans. Graph*, vol. 16, no. 1, pp. 34–73, 1997.

[67] Luo, R., Gilson, H., Potter, M., and Gilson, M., "The physical basis of nucleic acid base stacking in water," *Biophys. Journal*, vol. 80, pp. 140–148, 2001.

[68] Maillot, J., Yahia, H., and Verroust, A., "Interactive texture mapping," pp. 27–34, 1993.

[69] Malfait, M. and Roose, D., "Wavelet-based image denoising using a markov random field a priori model," *IEEE Trans. Image Proc.*, vol. 6 (4), pp. 549–565, 1997.

[70] M.Jansen, Nason, G., and Silverman, B., "Scattered data smoothing by empirical bayesian shrinkage of second generation wavelet coefficients," in *SPIE Vol. 4478*, pp. 87–97, 2001.

[71] Moore, P., "Structural motifs in RNA," *Annu. Rev. Biochem.*, vol. 68, pp. 287–300, 1999.

[72] Munkres, J., *Elements of Algebraic Topology.* Addison-Wesley Co., 1984.

[73] Murray, L., Arendall, W., Richardson, D., and Richardson, J., "RNA backbone is rotameric," *PNAS*, vol. 100, pp. 13904–13909, 2003.

[74] Nain, D., Haker, S., Bobick, A., and Tannenbaum, A., "Multiscale 3d shape analysis using spherical wavelets," in *Proc. of MICCAI*, 2005.

[75] Nason, G., "Choice of the threshold parameter in wavelet function estimation," *Wavelets and Statist., Lect. Notes in Statist.*, 1994.

[76] Noller, H., "RNA structure: Reading the ribosome," RNA, vol. 309, pp. 1508–1514, 2005.

[77] Ogden, T. and Parzen, E., "Change-point approach to data analytic wavelet thresholding," *Statist. Comput.*, vol. 6, pp. 93–99, 1996.

[78] Ogden, T. and Parzen, E., "Data dependent wavelet thresholding in non-parametric regression with change-point applications," *Comput. Statist. Data. Anal.*, vol. 22, pp. 53–70, 1996.

[79] Pizurica, A., Philips, W., Lemahieu, I., and Acheroy, M., "A joint inter- and intrascale statistical model for bayesian wavelet based image denoising," *IEEE Trans. on Image Processing*, vol. 11, pp. 545–557, 2002.

[80] Richardson, J., Schneider, B., Murray, L., Kapral, G., Immormino, R., Headd, J., Richardson, D., Ham, D., Hershkovits, E., Williams, L., Keating, K., Pyle, A., Micallef, D., Westbrook, J., Helen, M., and Berman, H., "RNA backbone: Consensus all-angle conformers and modular string nomenclature," RNA, p. to be published, 2008.

[81] Ruggeri, F. and Vidakovic, B., "A bayesian decision theoretic approach to wavelet thresholding," *Stat. Sinica*, vol. 9 (1), pp. 183–197, 1999.

[82] Saenger, W., *Principles of Nucleic Acid Structure.* Springer-Verlag, New York, 1984.

[83] Sander, P., Snyder, J., Gortler, S., and Hoppe, H., "Texture mapping progressive meshes," pp. 409–416, 2001.

[84] Sarver, M., Zirbel, C., Stombaugh, J., Mokdad, A., and Leontis, N., "Fr3d: Finding local and composite recurrent structural motifs in RNA 3d structures," *J. of Math. Biol.*, vol. 56, pp. 215–252, 2008.

[85] Schneider, B., Moravek, Z., and Berman, H., "RNA conformational classes," *Nucleic Acids Research*, vol. 32, pp. 1666–1677, 2004.

[86] Schreiner, J., Asirvatham, A., Praun, E., and Hoppe, H., "Inter-surface mapping," *ACM Trans. on Graph.*, vol. 23(3), pp. 867–874, 2004.

[87] Schröder, P. and Sweldens, W., "Spherical wavelets: Efficiently representing functions on the sphere," in *SIGGRAPH '95 Conference Proc.*, pp. 161–172, 1995.

[88] Schröder, W., Zarge, J., and Lorensen, W., "Decimation of triangle meshes," in *SIGGRAPH '92 Conference Proc.*, pp. 65–70, 1992.

[89] Sethian, J., *Level-Set Methods: Evolving Interfaces in Geometry, Fluid Dynamics, Computer Vision, and Material Science*. Cambridge Monograph on Applied and Computational Mathematics, Cambridge University Press, 1996.

[90] Simoncelli, E. and Adelson, E. H., "Noise removal via bayesian wavelet coding," pp. 379–382, 1996.

[91] Sponer, J. and Lankas, F., *Computational Studies of* RNA *and* DNA. Dordrecht, Netherlands, 2006.

[92] Sponer, J., Leszczynski, J., and Hobza, P., "On the nature of nucleic acid base stacking. nonempirical ab initio and empirical potential characterization of 10 stacked base pairs. comparison of stacked and h-bonded base pairs," *J. Phys. Chem.*, vol. 100, pp. 5590–5596, 1996.

[93] Sweldens, W., "The lifting scheme: a construction of second generation wavelets," *SIAM J. on Math. Anal.*, vol. 29, pp. 511–546, March 1998.

[94] Sykes, M. and Levitt, M., "Describing RNA structure by libraries of clustered nucleotide doublets," *Journal of Molecular Biology*, vol. 351, pp. 26–38, 2005.

[95] Taubin, G., "Estimating the tensor of curvature of a surface from a polyhedral approximation," in *Proc. of ICCV*, 1995.

[96] Taubin, G., "A signal processing approach to fair surface design," in *SIGGRAPH '95 Conference Proc.*, pp. 351–358, 1995.

[97] Turk, G., "Re-tiling polygonal surfaces," in *SIGGRAPH '92 Conference Proc.*, pp. 55–64, 1992.

[98] Vidakovic, B., "Nonlinear wavelet shrinkage with bayes rules and bayes factors," *J. of the Amer. Stat. Assoc.*, vol. 93, pp. 173–179, 1998.

[99] Vollmer, J. and Muller, H., "Improved laplacian smoothing of noisy surface meshes," in *Proc. of Eurographics*, pp. 131–138, 1999.

[100] Waller, M., Robertazzi, A., Platts, J., Hibbs, D., and Williams, P., "Hybrid density functional theory for π-stacking interactions: Application to benzenes, pyridines, and dna bases," *Journal of Comput. Chem.*, vol. 27, pp. 491–504, 2006.

[101] Wang, J. and Swendsen, R., "Cluster monte carlo method," *Physica A*, vol. 1:167, pp. 565–579, 1990.

[102] Weyrich, N. and Warhola, G., "Wavelet shrinkage and generalized cross validation for image denoising," *IEEE Trans. Image Proc.*, vol. 7, pp. 82–90, 1998.

[103] Zhu, L., Haker, S., and Tannenbaum, A., "Flattening maps for the visualization of multibranched vessels," *IEEE Trans. Med. Imag.*, vol. 24(2), pp. 191–198, 2005.